Universidad Miguel Hernández

Departamento de Ingeniería
de Sistemas y Automática

Perceptual image coding for wavelet based encoders.

Tesis Doctoral

Memoria presentada para optar
al grado de Doctor por:
Miguel Onofre Martínez Rach

dirigida por:
Manuel Pérez Malumbres
Otoniel Mario López Granado

Universidad Miguel Hernández

Departamento de Ingeniería
de Sistemas y Automática

Perceptual image coding for wavelet based encoders.

PhD thesis

A dissertation for the degree of
Doctor of Philosophy by:
Miguel Onofre Martínez Rach

Advisors:
Manuel Pérez Malumbres
Otoniel Mario López Granado

D. MANUEL PÉREZ MALUMBRES, Catedrático de Universidad de la Universidad Miguel Hernández y D. OTONIEL MARIO LÓPEZ GRANADO, Profesor Contratado Doctor de la de la Universidad Miguel Hernández,

CERTIFICAN:

Que la presente memoria *Perceptual image coding for wavelet based encoders*, ha sido realizada bajo su dirección, en el Departamento de Ingeniería de Sistemas y Automática de la Universidad Miguel Hernández, por el Ingeniero D. MIGUEL ONOFRE MARTÍNEZ RACH, y constituye su tesis para optar al grado de Doctor.

Para que conste, en cumplimiento de la legislación vigente, autorizan la presentación de la referida tesis doctoral ante la Comisión de Doctorado de la Universidad Miguel Hernández, firmando el presente certificado.

Elche, 3 de Noviembre de 2014

Fdo. Manuel Pérez Malumbres Otoniel Mario López Granado

D. JOSE MARÍA AZORÍN POVEDA, Profesor Titular de Universidad y director del Departamento de Ingeniería de Sistemas y Automática de la Universidad Miguel Hernández,

CERTIFICAN:

Que la presente memoria *Perceptual image coding for wavelet based encoders*, realizada bajo la dirección de D. MANUEL PÉREZ MALUMBRES y D. OTONIEL MARIO LÓPEZ GRANADO, en el Departamento de Ingeniería de Sistemas y Automática de la Universidad Miguel Hernández, por el Ingeniero D. MIGUEL ONOFRE MARTÍNEZ RACH, constituye su tesis para optar al grado de Doctor.

Para que conste, en cumplimiento de la legislación vigente, autoriza la presentación de la referida tesis doctoral ante la Comisión de Doctorado de la Universidad Miguel Hernández, firmando el presente certificado.

Elche, 3 de Noviembre de 2014

Fdo. José María Azorín Poveda

Acknowledgements

A mis padres, hermana e hijos

Quiero dedicar este trabajo a mi familia, por el cariño y apoyo que siempre me han dado. A mis padres por la confianza que siempre han depositado en mí, por su ejemplo y dedicación continua en nuestra educación.

Quiero agradecer en primer lugar y especialmente a mis directores de tesis y amigos Manuel y Otoniel, que me han dedicado mucho tiempo, esfuerzo y paciencia para guiarme y ayudarme, sobre todo en los momentos difíciles durante el desarrollo de esta tesis doctoral.

También quiero agradecer a todos los compañeros que me habéis ayudado y colaborado en las publicaciones que soportan esta tesis; Pablo Piñol, Vicente Galiano, Hector Migallón, Jesús LLor, Estefanía Alcocer, Carlos Calafate y José Oliver, a todos muchas gracias por vuestra amistad, por las horas de trabajo y debate, por vuestra paciencia y escucha y por vuestro apoyo y sabios consejos.

Agradecer también al Dr. Anil Fernando y a todo el equipo del I-Lab Multimedia and DSP research group de la Universidad de Surrey en el Reino Unido por facilitar y ayudarme durante mi estancia pre-doctoral en dicha universidad.

Agradecer finalmente, a mi universidad y a los departamentos de Física y Arquitectura de computadores y de Ingeniería de Sistemas y Automática por el soporte que me han dado, y en general quiero agradecer a todos los que de una forma u otra habéis hecho posible que este trabajo llegara a buen término.

Gracias a todos.

II

Abstract

In the literature, we can find numerous works in the fields of image and video coding and compression that present new algorithms, techniques, or methods that lead to some improvements in comparison to previous proposals or standards. Some improvements are related with complexity reduction of encoding/decoding engines, others with the use of resources such as memory or computational cost, and others with the quality of the reconstructed image or sequence, i.e., obtaining the same quality with less compression rate or more quality at the same rate.

Once a new proposal is developed, it must compare its results with previous proposals or standards in order to quantify the gains. Such comparisons should be done not only at the end of the proposal development, but also during the design stages. Researchers in these fields are continuously comparing the results obtained after any modification in the algorithms or methods, with the results of the reference encoder, in order to tune or guide their research to obtain higher gains in coding time, resource usage, and/or quality.

Focusing on quality comparisons, i.e., quality comparisons at a specific bit rate or rate comparisons at the same quality, we see that researchers have adopted the Mean Square Error (MSE) and the Peak to Noise Ratio (PSNR) as a *de facto* standard metric to compare and measure the quality of the reconstructed images or videos. This is because MSE, and consequently PSNR, have many attractive features [1]: they are simple to calculate and parameter free, mathematically easy to deal for optimization purposes, are the natural way to define the energy of the error signal, and finally they are the most commonly used metrics. Technically, MSE measures image difference, whereas PSNR measures image fidelity. The main idea was, therefore, that reducing the mathematical error between images leads to a better quality of the reconstructed image.

Also, the use of Rate/Distortion (R/D) curves to compare and measure the behavior of the encoding proposals over a range of bit rates are widely used. From a designer's point of view, it is interesting to have a unique value that measures how good a proposal is with respect to others for a specific bit rate

range, and not only at a specific bit rate, and so recently [2], the Bjontegaard method to compare R/D curves was also adopted as the *de facto* standard for this need.

When the image reconstruction is not mathematically exact, i.e., the PSNR difference between the original and the reconstructed image is below a threshold, the perception of quality given by our Human Visual System (HVS) is in many cases far from the mathematical quality value given by the PSNR. As we will see later, there are many examples where the PSNR value for different reconstructed images with different distortions is almost the same; however, humans clearly perceive different qualities for each image, and can even rank those images by quality.

Therefore, there is a need for measuring the quality of reconstructed images and video sequences in a similar way as the HVS. As we will see later, there are many proposals, or contributions in this field trying to achieve this from different approaches. In this thesis, we review the most outstanding proposals in the field of Quality Assessment Metrics (QAM), and we will focus on the metrics for which the source code was available. We also review and discuss the methods that must be followed in order to fairly compare different metrics using a common quality scale.

The equation to translate a metric to this common quality scale needs some parameters that are seldom published, and as we will see later, there are many factors that produce variations in the comparison results, so, in this thesis, we will perform a comprehensive comparison and we publish the parameters that will translate each metric to that common scale. The comparison sets the degree of correlation of a metric with the subjective quality assessment. In this work, we will also analyze how the metrics behave in different environments, specifically in the image and video compression environment, and in mobile networks where packet losses are present. This study draws conclusions about which metric adapts better to each environment.

As previously commented, the PSNR became a standard before research in the QAM field provided the metrics, but even after, when some metrics were available, and widely recognized as having a better correlation with subjective quality assessment than PSNR, only a few works have used them to compare their quality performance results.

In this thesis, we also review the most important and widely used techniques to provide perceptual improvements in image and video encoders. As many of the proposals include these techniques in their algorithms, in some of the encoding stages it is reasonable to measure the performance of such proposals from a perceptual point of view, i.e., using a QAM in the comparisons instead of the PSNR. But nowadays, the PSNR is still predominantly used, comparing at a specific rate or using PSNR R/D curves, i.e., R/D curves where the distortion metric is PSNR. Recently, however, there have been some works that include,

besides PSNR, other QAM in their performance comparisons.

We will also include the most relevant perceptual techniques into the S-LTW encoder, a non embedded wavelet based encoder, in order to obtain a new perceptual enhanced encoder that we call PETW (Perceptually Enhanced Tree Wavelet). The techniques that we will explore and include in the new encoder can also be used in other wavelet- or even DCT-based encoders with the corresponding modifications. As our new encoder includes perceptual techniques in its design, we will use the VIF QAM, which was the one that obtained the best correlation in our metrics comparison tests.

The most widely used perceptual technique is the adaptation of the HVS sensitivity to contrast by means of the Contrast Sensitivity Function (CSF) into the quantizer. Many authors perform complex subjective tests in order to fix the perceptual importance of each frequency band or subband, and so, include the HVS sensitivity to contrast in the encoder using empirical weighting matrices. Some other authors obtain the weighting matrix directly from a model of the CSF. In order to avoid the need to perform that complex subjective tests, in this thesis we will deeply analyze this technique so we can improve the way in that the weighting matrix is obtained, and therefore improve the perceptual performance of the encoder when using our proposal. This performance comparison is also made in terms of VIF R/D analysis against the reference work and using well-known image and video encoders (in intra mode).

Finally, and inspired by some studies from the quantization research field, we designed an adaptive estimator of the dead zone size in our new encoder so that the perceptual quality of each reconstructed image is optimized. In order to be able to use this dead zone variability, we replace the original quantization stages of the S-LTW with a Uniform Variable Dead Zone Quantizer (UVDZQ). Using this quantizer jointly with our dead zone adaptive estimator and the perceptual weighting matrices, we improve the perceptual R/D behavior of the S-LTW encoder. This new encoder is also compared with other well-known image encoders with and without perceptual enhancements.

Our comparisons results determine that the union of the techniques used in this thesis achieve better perceptual R/D behavior of the PETW encoder against the reference S-LTW and other well-known encoders, with great bit rate savings when encoding an image with the same perceptual quality.

VI

Resumen

En el ámbito de la codificación y compresión de imagen y vídeo son numerosas las aportaciones que encontramos en la literatura que presentan mejoras sobre anteriores o sobre los estándares en algún aspecto. Unas plantean diferentes o nuevas formas de codificar, otras refinan los métodos ya existentes, para finalmente tratar de mejorar el rendimiento de la propuesta sobre las anteriores, bien en tiempo de codificación, utilización de recursos, tasa de compresión o calidad de la imagen o el video reconstruido, etc...

Una vez desarrollada una nueva propuesta la gran mayoría necesita comparar sus resultados con los de propuestas anteriores. Pero estas comparaciones no solo se producen cuando se tiene el nuevo codificador o un nuevo método completamente terminado, sino que lo que es más costoso, es la necesidad de recurrir continuamente a comparativas durante el tiempo de diseño en cualquiera de sus aspectos, tiempo de procesamiento, memoria, calidad, etc. para modificiar los métodos y lograr el mayor rendimiento.

Si nos centramos en comparativas de calidad a una misma tasa de bits o bien en comparativas de tasa de bits para una misma calidad, vemos cómo la comunidad científica ha adoptado historicamente el MSE (Mean Square Error) y el PSNR (Peak to Noise Ratio) para medir la calidad y analizar los rendimientos en Rate/Distortion (R/D) de las distintas propuestas, pues posee cualidades muy atractivas [1] ya que es simple de calcular, su formula no necesita parámetros, matemáticamente se puede utilizar en algoritmos de optimización, es la forma natural de definir la energía del error de la señal y por último es la métrica más utilizada por lo que permite las comparaciones. La idea básica e indiscutible que favoreció al PSNR es que la mejor imagen reconstruida es la que es matemáticamente idéntica a la original.

Desde el punto de vista del diseñador de un codificador lo interesante es determinar cuan buena es la propuesta para uno o varios rangos de rate, no únicamente para un rate determinado, pero sin que para ello sea necesario codificar y decodificar a todaas las tasas. Por ello surgió la propuesta de Bjontegaard que propone un único valor porcentual de la mejora de una propuesta respecto a otra basándose en el comportamiento R/D en PSNR de ambas y que ha sido adoptada como estandar de facto.

Cuando la recuperación de una imagen reconstruida ya no es matemáticamente exacta, la percepción visual de la calidad por el sistema visual humano dista mucho en ciertos casos de lo que la comparación matemática dicta como valor de calidad. Como veremos más adelante son muchos los ejemplos en los que la comparación entre distintas imagenes reconstruidas, provenientes de distintas distorsiones, contra un mismo original resulta en un mismo valor de calidad PSNR, pero el sistema visual humano determina claramente diferentes calidades y una ordenación diferente en cuanto a calidad.

Por lo tanto surgió la necesidad de poder medir la calidad de las imágenes y videos reconstruidos de una forma más parecida a cómo lo hace nuestro sistema visual. Como veremos son muchas las aportaciones en este ámbito y desde muchas aproximaciones diferentes. En esta tesis realizamos una revisión de las aportaciones más relevantes en éste ámbito y nos centraremos en analizar el comportamiento de aquellas para las que dispusimos de su código fuente. A su vez revisamos y discutimos la metodología para poder comparar métricas entre si utilizando una escala común.

La ecuación que permite trasladar una metrica a esta escala común requiere de unos parámetros que no suelen publicarse en la literatura y como veremos posteriormente son muchos los factores que hacen variar los resultados de las comparativas, por tnato en esta tesis nosotros realizamos la comparativa completa publicando los parametros obtenidos. Esta comparativa determina el grado de correlación de una metrica a la valoración subjetiva de calidad. En este trabajo además se analiza cómo se comportan estas métricas en varios escenarios, en concreto, cómo responden las métricas frente a resultados de compresión y frente a la pérdida de paquetes en redes móviles. Este estudio arroja conclusiones sobre qué métrica se adapta mejor a que tipos de compresión y perdida.

Como hemos comentado, el PSNR se convirtió en estandar de facto antes de que se pudieran utilizar métricas de calidad perceptual pero aun existiendo éstas, posteriormente son pocos los trabajos que comparan su rendimiento utilizando métricas perceptuales a pesar de que está ampliamente reconocido que tienen mejor correlación con la valoración subjetiva de calidad que el PSNR.

En este trabajo también se revisan las técnicas perceptuales más importantes que se utilizan para incorporar aspectos perceptuales en la codificación de imagen y video. Muchas son las propuestas que incluyen estas técnicas en la codificación de imagen y video. Al incluir algoritmos perceptuales en alguna de las etapas de un codificador o de una propuesta de mejora de éstos, lo razonable es medir el rendimiento de la propuesta desde un punto de vista perceptual. Pero aún, mayoritariamente los trabajos siguen utilizando el PSNR o las curvas R/D donde el PSNR es la métrica de calidad,

aunque últimamente algunos trabajos comienzan a utilizar, además del PSNR, otras métricas de calidad perceptual.

En este trabajo incluiremos técnicas perceptuales de codificación en un codificador wavelet no embedido, el S-LTW, para realizar una propuesta de un nuevo codificador, el PETW (Perceptually Enhanced Tree Wavelet). Muchas de las técnicas que aqui exploramos y proponemos, pueden ser extrapoladas a otros codificadores basados en la transformada wavelet o en la DCT. Puesto que el codificador incluye elementos perceptuales, todas nuestras comparaciones de rendimiento se realizan utilizando la métrica perceptual VIF, que resultó la más correlacionada con la valoración subjetiva en las comparaciones realizadas.

La técnica perceptual más extendida en la codificación perceptual es la inclusión mediante la CSF (Contrast Sensitivity Function) de la sensibilidad al contraste del sistema visual humano en la etapa de cuantización. Muchos autores realizan test subjetivos con el fin de determinar la importancia perceptual de cada banda o subbanda de frecuencia y así incluir la sensibilidad al contraste del HVS en los codificadores gracias a unas matrices de pesos empiricamente obtenidas. Otros autores, sin embargo, obtienen las matrices de pesos directamente de un modelo de la CSF. Con el fin de evitar los costosos test subjetivos, nosotros analizaremos esta técnica y propondremos mejoras en la selección de pesos y mejorar por tanto el rendimiento del codificador al usar nuestra matriz de pesos. Nuestra propuesta será contrastada en terminos R/D usando la VIF como métrica de calidad comparandola con la referencia y utilizando varios codificadores de imagen y video en modo intra en las comparativas.

Por último, y basado en resultados de varios estudios, incluimos en nuestro codificador un estimador adaptativo que en función de la imágen estima un valor óptimo del ancho del dead zone para el cuantizador utilizado. Por ello modificamos la cuantización original de S-LTW para sustituirlo por un UVDZQ (Uniform Variable Dead Zone Quantizer). Este estimador determina el ancho del dead zone que permitirá mejorar el rendimiento R/D perceptual respecto a usar el dead zone que usaba el S-LTW. El rendimiento de este estimador adaptativo se compara con otros codificadores de imagen muy conocidos.

Nuestras comparaciones determinan que la unión de las técnicas utilizadas en esta tesis consiguen mejorar el comportamiento perceptual R/D de nuestro codificador frente al S-LTW y a otros codificadores, y conseguir considerables ahorros en rate al codificar una imagen a una misma calidad perceptual, es decir, para una misma calidad perceptual el PETW consigue reducir la tasa de bits utilizada.

Preface

Motivation

The main goal is to propose a new perceptually driven image encoder, but for this task we have to deal with some issues that arose when we began this task and that motivate this work. As starting point for our new encoder we chose the S-LTW encoder [3], a non embedded wavelet based encoder. As obtaining better results in terms of perceptual quality, drives the design and adaptations in our encoder, it is evident that we need to measure the obtained quality from a perceptual point of view and not using the traditional PSNR metric.

When performing the *state-of-the-art* review of the perceptual coding techniques, we observe that most of the works did not use perceptual QAM. They mainly use printed images encoded at a specific bit rate for visual inspection of the benefits of the applied technique. Only few works use QAM while others use also PSNR R/D curves. While working with some perceptual techniques in our encoder, we observe also that measuring the R/D performance with PSNR as distortion metric will produce misleading interpretations of the suitability of these perceptual coding techniques in that encoder, as the R/D curve of the perceptually modified encoder gets worse results than the original one. This happen because these techniques change the perceptual importance of the wavelet coefficients, quantizing them in a different way. This produces higher mathematical differences in the reconstruction image that PSNR detects as fidelity errors, but a visual inspection of that images reveals that quality is even higher or that for the same perceptual quality a bit rate reduction is achieved.

In order to measure the perceptual quality performance of our encoder we could use subjective tests. But this is a very cumbersome, and time and resource consuming task. More over, in the design stage, were each time that a modification in the algorithms is done, a new test must be run in order to detect if that modification works as expected. So, one motivation is to avoid the continuous subjective tests to verify each design decision. While reviewing the perceptual coding techniques used in wavelet based encoders, we see that some techniques avoid the use of subjective test to determine the perceptual

importance of the wavelet coefficients, so we decide to study it deeply and try to improve its results.

So, on the one hand we know that we have to measure the performance from a perceptually point of view and on the other one we want to avoid the use of continuous subjective tests. Therefore using a QAM for this task is another motivation in this work, but the question is, which one? In order to answer this question, we need a review in that research field. Several issues, as we will explain later, forced us to perform an ad-hoc QAM comparison, and as one of the future uses of our encoder is to send compressed images over mobile networks, we have to analyze also the behavior of the metrics in that environments, i.e., compression and mobile networks with packet losses. Once the metric has been selected, every performance comparison should be done with it.

Finally, another question arises while observing that most of the perceptual coding techniques are non image adaptive, and that the existing adaptive techniques use to have high computational costs, or use some global parameters that are valid for a set of images but not for each specific one. In a review of perceptual quantization techniques we observe that some authors performed studies that were focused on obtain a better PSNR R/D performance by modifying the dead zone size when a dead zone quantizer is used. So our motivation was finally to design an image adaptive estimator of the dead zone size that improves the perceptual R/D performance with lower computational cost and easy to implement in the encoding engine.

Objectives

The specific objectives of this thesis can be detailed as follows:

- Study the state-of-the-art in the field of objective perceptual quality assessment metrics
- Comparison of the most representative metrics, and evaluation of their behavior in compression and packet losses environments.
- Study the state-of-the-art of the perceptual coding techniques and determine which ones can be included in our wavelet non-embedded image encoder.
- Improve the performance results of that perceptual techniques.
- Use a wavelet-based intra encoder as reference where the most relevant perceptual coding techniques will be implemented.
  - Add the improved perceptual coding techniques.

– Study the impact of the dead zone size in the final reconstructed perceptual quality.

Thesis organization

This thesis is organized in four chapters, which are introduced here:

Chapter 1 presents the fundamentals of image and video compression. In Section 1.1.2, we present several state-of-the-art wavelet image encoders, emphasizing non-embedded encoders. The present image coding standard JPEG 2000 is also presented in Section 1.1.3. On the other hand, in Section 1.2.2 and 1.2.4, both the video coding standard H.264 (also referred to as Moving Picture Experts Group (MPEG)-4, part-10) and other video encoders are presented.

In Chapter 2 a study of the most commonly produced artifacts in image and video coding is presented jointly with an overview of the Human Visual System characteristics referred in this work. In sections 2.4 and 2.5 an in depth objective perceptual quality assessment metrics study and classification are presented. In Section 2.6 a review and discussion of how to compare QAM is done, and finally in Section 2.7 some conclusions of this Chapter are summarized.

Chapter 3 begins in Section 3.1with a brief review of the basics quantization. In Section 3.2.1an in-depth perceptual coding study is presented where the most relevant techniques in perceptual coding are exposed. In Section 3.3 we analyze, propose and compare the performance of our perceptual subband weighting matrix. In Section 3.4the PETW encoder is introduced analyzing and discussing the use of the UVDZQ, so in Section 3.4.2 a performance evaluation of the PETW version with perceptual weights and the new quantizer is shown. This performance comparison is performed with other video encoders running in intra mode. And finally in section 3.4.3 we introduce the new image adaptive dead zone estimator that improves the perceptual R/D performance. The performance of this estimator is evaluated in Section 3.4.4.

Finally, Chapter 4 concludes and summarizes some of the main contributions introduced in this thesis. We also advance here some of the future research.

XIV

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Contents

1.1 Image coding

Compression of digital images plays a key role in image storage and transmission. In this chapter a brief introduction to general image compression will be given.

1.1.1 Fundamentals

The usefulness of digital images in information transmission is not questionable, but the cost of storing and transmitting images is much larger compared to storage and transmission of text, so that for example image databases require more storage than document archives.

The amount of data transmitted via the Internet doubles every year, and a large portion of that data are images and video sequences. Reducing the bandwidth needs of any given device will result in significant cost reductions and will make the device more affordable. Magnetic hard discs (HD)s, CDs, DVDs, and Solid State Drives (SSD)s of larger capacity are released every year, in response to greater demand for storage of digital data. Image compression offers ways to represent an image in a more compact way, so that one can store more images and transmit images faster. The advantages of image compression come at the expense of a computational cost. Before storing or transmitting an image it is processed in such a way that will require fewer bits to represent it.

A compression algorithm tries to offer the best trade-off between the bandwidth, memory, computation factors and quality for a given application. For example, if we are limited in terms of memory we can spend more computational time to compress the image and make sure it fits into the given memory size. If we are computation limited we can store the image as it is with no compression or with limited compression with a simple compression algorithm.

Image compression algorithms have been the subject of research both in academia and industry for many years, but there is still room for new technologies. The first widely adopted international image compression standard was Joint Photographic Experts Group (JPEG) [5, 6] which was introduced in the late eighties. JPEG is based on Discrete Cosine Transform (DCT) followed by entropy coding based on either Huffman coding [7, 8, 9] or binary arithmetic coding [10, 11, 9, 12]. It has been widely used from the printing industry to Internet applications. For example, all high-end printers compress the image to be printed before they actually send it to the print engine, and most images transmitted via the internet are JPEG compressed. JPEG is intended for continuous tone images of more than one

bit depth. Algorithms for binary images work in a different way, and JBIG-1 and JBIG-2 are the standards covering this area. There are other standards, such as facsimile transmission standards [13], the FlashPix file format [14], the TIFF file format [15], and page description languages like Probability Density Function (PDF).

There are two major classes of image compression algorithms, namely lossy and lossless algorithms. Lossless algorithms preserve the image data, i.e. original and reconstructed images are exactly the same. In lossy image compression, original and reconstructed images may or may not be identical in a strict mathematical sense, but to a human observer they may look the same, so the goal is to achieve compression that is visually lossless. Both lossy and lossless compression algorithms are used today in a broad range of applications, from transmitting satellite images, to web browsing to image printing and scanning. With lossy compression algorithms we can achieve significantly larger compression ratios compared to lossless algorithms.

1.1.1.1 Generic compression system

Most image coders consist of transform, quantization and entropy coding, as seen in Figure 1.1. The transform block is in general a reversible operation, i.e. a cascade of forward and inverse transform block is the identity operation. $T.T^{-1}(arg) = T^{-1}.T(arg) = arg$. Quantization, on the other hand, introduces some loss. The quantizer usually maps an interval of real numbers to a single index, constituting the only lossy part of the coding system i.e., $Q^{-1}.Q(arg) \neq arg$. It is lossy because the knowledge of an index is only enough to give us the corresponding interval in the real line but not the exact number in the real line. The entropy coder is the building block responsible for compression, it maps more frequent indexes to small codewords and less frequent indexes to larger codewords. It is also a reversible operation. A large portion of the computational complexity of a compression system is due to the entropy coding part of the system. More compression usually translates to higher computational complexity. In general, arithmetic [10] and Huffman coding [7] are the most common choices. Arithmetic coding is intended for high-end applications where complexity is not a concern, but compression performance is, while Huffman coding is intended for low-end applications where simplicity is more important. Typically the most memory intensive element is the transform. Quantization, on the other hand, is a much simpler process than the transform or the entropy coder.

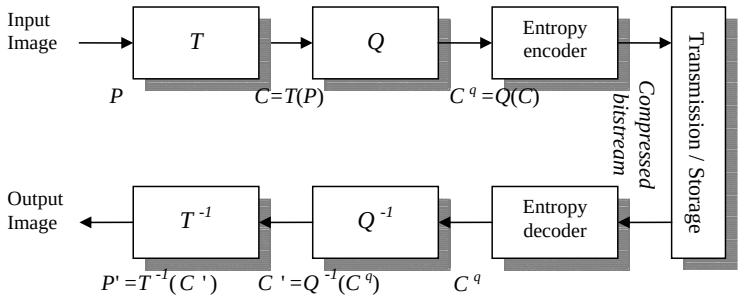


Figure 1.1: Overview of an image coder and decoder based on transform coding. T and T^{-1} are the forward and inverse transform functions, respectively. Q and Q^{-1} are the quantizer and dequantizer functions, respectively. The original set of pixels is represented by P

1.1.2 Wavelet based encoders

The wavelet transform is able to spatially decorrelate the image pixels in a linear way. However, more complex dependencies exist in natural images. Therefore, we still need good processing techniques, beyond simple entropy coding, in order to reduce these high-order statistical dependencies and so improve compression efficiency. The way in which wavelet coefficients are encoded establishes the coding model and it is the main difference among different encoders. In this section, we survey some of the most important wavelet-based image encoders that have been reported in the literature. In the performance analysis of each proposal, we not only focus on their coding efficiency but also on their complexity, since reduced complexity is one of the objectives of this thesis.

1.1.2.1 Overview

The wavelet transform computation represents only the first step in transform coding, and it is employed to decorrelate the input samples (pixels in the case of image coding), achieving a less redundant smaller area of coefficients, which concentrates most energy, whereas the remaining coefficients are reduced and, in many cases, become zero or very close to zero. Therefore, the Discrete Wavelet Transform (DWT) is a common point in wavelet coding, and there is almost no difference in this part from one wavelet-based encoder to another one. In this step, almost the only degree of freedom for an encoder is the wavelet family and the type of wavelet decomposition. Although most schemes are

based on the B9/7 transform [16] with a dyadic decomposition, other wavelet families and wavelet decompositions (such as wavelet packets [17, 18]) have been employed [19, 20, 21, 22].

Following the scheme depicted in Figure 1.1, after the DWT computation an encoder must define the way in which rate/distortion is modified through quantization, and how to encode the quantized coefficients. The way quantization and coding is applied defines a specific model for each wavelet encoder, and it is probably the main difference among different encoders.

Some wavelet encoders apply in combination the quantization and entropy coding steps, so as to improve coding performance by means of optimization algorithms (such as the Lagrange multiplier method [23, 18]), or to allow other features, like Signal to Noise Ratio (SNR) scalability (e.g., by applying quantization through successive approximation [24, 25]). Actually, the model employed not only establishes the compression performance but also other additional features of the output bit-stream. E.g., generally speaking, depending on the order in which coefficients are encoded, an image can be decoded with resolution or quality scalability.

A wide variety of wavelet-based image compression schemes have been reported in the literature, ranging from simple entropy coding to more complex techniques such as vector quantization [26, 27], tree-based coding [24, 25], block-based coding [28, 29], edge-based coding [30], joint space-frequency quantization schemes [31, 19], trellis coding [32], etc.

The early wavelet-based image coders [33, 16] were designed in order to exploit the ability of the wavelet transform to compact the energy of an image in a simple way. They employed scalar or vector quantizers and variable-length entropy coding, showing little improvement with respect to popular DCT-based algorithms like JPEG. In fact, in [34], some early wavelet encoders were compared with JPEG, concluding that these encoders obtained better results than JPEG only when very low bit rates were used (below 0.25 bits per pixel (bpp) for an original grey-scale 8 bpp image). However, despite a not very brilliant beginning, DWT has been successfully employed later in the field of image coding.

In this chapter, some of the most relevant and efficient wavelet tree-based coding techniques that have been proposed recently are surveyed. Among the wide variety of efficient encoders available in the literature, we highlight the non-embedded proposals and the fastest coding/decoding schemes. The reason why we focus on this type of encoder is that we are interested in models with low computational requirements.

1.1.2.2 Embedded zero-tree wavelet coding

In the early 90s, there was the general idea that more efficient image coding would only be achieved by means of sophisticated techniques with high complexity. The embedded zero-tree wavelet encoder (Embedded Zero-tree Wavelet (EZW)) [24] can be considered the first wavelet image coder that broke that trend. This encoder exploits the properties of the wavelet coefficients more efficiently than the rest of early techniques and thereby, it considerably outperforms their coding performance.

The EZW algorithm is mainly based on two basic ideas: (a) the similarity between the same type of wavelet subband, with higher energy as the subband level increases, and (b) a type of quantization based on a successive-approximation scheme that can be adjusted in order to get a specific bit rate in an embedded way. The former idea is exploited by means of coefficient trees, whereas the latter is usually implemented with bit-plane coding. In addition, the encoder includes an adaptive arithmetic encoder to encode the generated symbols. Although the EZW technique never became a standard, it is of great historical importance in the field of wavelet-based image coding because the aforementioned two principles were later used and refined by many other coding methods.

Let us define the coefficient trees employed in EZW. In a dyadic wavelet decomposition there are coefficients from different subbands representing the same spatial location in the sense that one coefficient in a scale corresponds spatially with four coefficients in the correspondent previous subband. This connection can be extended recursively with these four coefficients and the corresponding direct descendants (sometimes called offspring) at the previous levels, so that coefficient trees can be defined as shown in Figure 1.2. Since each node in a tree has four direct descendants (except the coefficients at the first level, corresponding with the leaf nodes), this type of tree is sometimes called quadtree. Note that a quadtree (or subquadtree) can be built from each coefficient by considering it as the root node of a tree.

The key idea employed by EZW to perform tree-based coding is that, in natural images, most energy tends to concentrate at coarser scales (i.e., higher decomposition levels). Then, it can be expected that the closer to the root node a coefficient is, the larger magnitude it has. Therefore, if a node of a coefficient tree is lower than a threshold, its descendant coefficients are likely to be lower as well. In other words, the probability for all four children to be lower than a threshold is much higher if the parent is also lower than that threshold. We can take advantage of this fact by coding the subband coefficients by means of trees and successive approximation, so that when a node and all its descendant

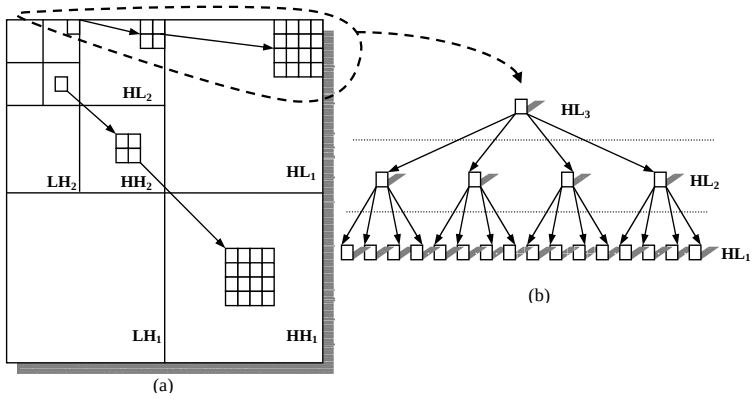


Figure 1.2: Definition of wavelet coefficient trees. In (a), it is shown that coefficients of the same type of subband (HL, LH or HH) representing the same image area through different levels can be logically arranged as a quadtree, in which each node is a wavelet coefficient. The parent/child relation between each pair of nodes in the quadtree is presented in (b)

coefficients are lower than a threshold, just a symbol is used to encode that entire branch.

The EZW algorithm is performed in several steps, with two stages per step: the dominant pass and the subordinate pass. Successive-approximation can be implemented as a bit-plane encoder so that the method can be outlined as follows: Consider that we need n bits to represent the highest coefficient of the image (in absolute value). Then, the first step will be focused on all those coefficients that need exactly n bits to be coded (ranging from 2^{n-1} to $2^n - 1$), which are considered to be significant with respect to n. In the dominant pass, each coefficient falling in this range (in absolute value) is labeled and encoded as significant positive/negative (*sp*/sn), depending on its sign. These coefficients will no longer be processed in further dominant passes, but in subordinate passes. On the other hand, the remaining coefficients (those in the range $[0, 2^{n-1}[$) are encoded as zero-tree root (*zr*) if all its descendants also belong to this range, or as isolated zero (*iz*) if any descendant is significant. Note that no descendant of a zero-tree root needs to be encoded in this step, because they are already represented by the zero-tree root symbol. In the subordinate pass, the bit n of coefficients labeled as *sp*/*sn* in any prior step is coded. In the next step, the n value is decreased by one, so that we focus now on the following bit (from Most Significant Bit (MSB) to Least Significant Bit (LSB)). This compression process finishes when the desired bit rate is reached, and the decoder can partially use the incoming bit-stream to

reconstruct a progressively improved version of the original image. That is why this coder is called embedded.

In the dominant pass, four types of symbols need to be coded: *sp*, *sn*, *zr*, and *iz*, whereas in the subordinate pass only two are needed (bit zero and bit one). In order to get higher compression, an adaptive arithmetic encoder is used to encode the symbols computed during the dominant pass.

Due to its successive-approximation nature, EZW is SNR scalable, although at the expense of sacrificing spatial scalability. In addition, line-based wavelet transforms [35] are not suitable for this encoder, because the whole image is needed in memory to perform several image scans focusing on different bit planes and searching for zero-trees. Moreover, EZW needs to compute coefficient trees and performs multiple scans on the transform coefficients, which involves high computational time, most of all in cache-based architectures due to the higher cache miss rate.

1.1.2.3 Set partitioning in hierarchical trees

Said and Pearlman [25] proposed a variation of EZW, called SPIHT, which is able to achieve better results than EZW even without arithmetic coding. SPIHT is based on the same principles as EZW. However, improvements are mainly due to the way it searches for significant coefficients in the quadtrees, by splitting them with a novel partitioning algorithm.

Like in EZW, SPIHT encodes the wavelet subbands in successive steps, focusing on a different bit plane in each step. For a certain bit plane (n), the set partitioning sorting algorithm included in SPIHT identifies the insignificant coefficients in the transformed image. This algorithm encodes the coefficient significance by means of significance tests, which query each set to know if it has at least one significant coefficient. If so, it divides that set into more subsets and it then repeats the same question, otherwise we have identified a group of insignificant coefficients with respect to the current bit plane. The result of each query is encoded with a simple binary symbol, so that the decoder can reconstruct the same groups of insignificant sets. The subsets with significant coefficients are successively divided until each single significant coefficient is identified. When all the subsets are found to be insignificant with respect to the current bit plane, all the significant coefficients have been located, and the sorting pass is over for this step. The algorithm then encodes the corresponding bit (n) of those coefficients found significant in previous steps, which is called the refinement pass. Afterward, it focuses on the following bit plane ($n - 1$) and repeats the same process until the desired bit rate is reached. Note that the sorting and refinement passes of SPIHT are equivalent in concept to the

dominant and subordinate passes of EZW, respectively.

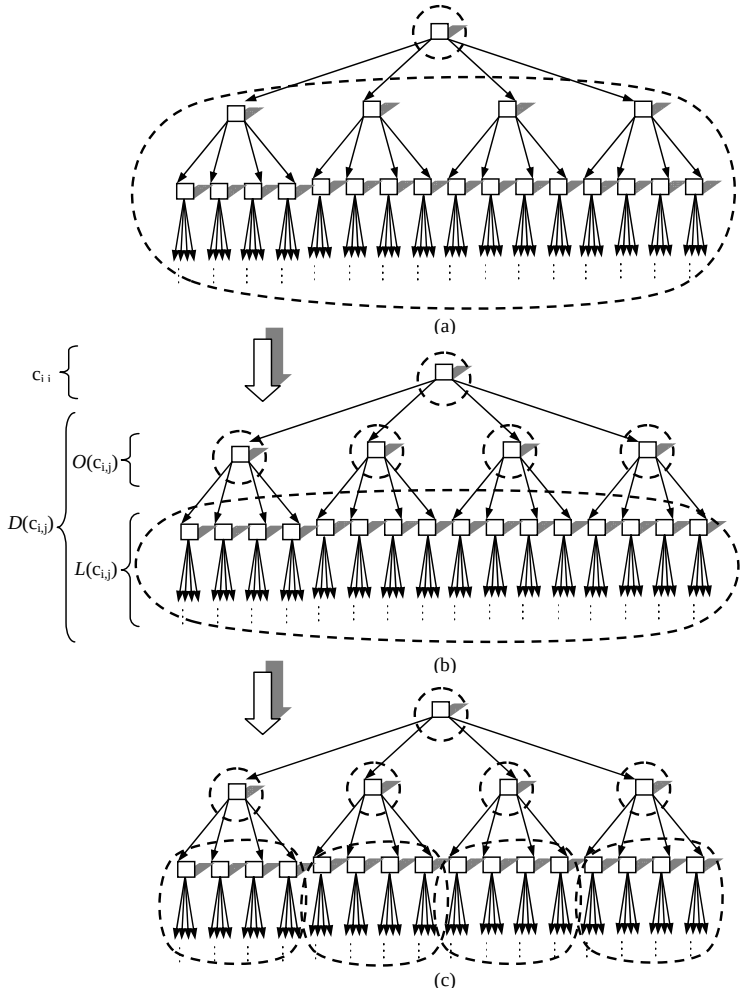


Figure 1.3: Example of division of coefficient sets arranged in spatial orientation trees. This division is carried out by the set partitioning sorting algorithm executed in the sorting pass of SPIHT. The descendants of $c_{i,j}$ presented in (a) are partitioned as shown in (b); if needed, the subset of (b) is divided as shown in (c), and so on

SPIHT uses spatial orientation trees (which are basically the quadtrees of Figure 1.2) to construct the initial set of coefficients and to establish the rules to divide them in the sorting algorithm. The notation employed in the algorithm is shown in Figure 1.3(b). For a given coefficient $c_{i,j}$, $D(c_{i,j})$ is the set of all the

descendant coefficients of $c_{i,j}$. This set can be split into direct descendants (or offspring) $O(c_{i,j})$ and non-direct descendants $L(c_{i,j})$.

In the SPIHT algorithm, the initial sets of coefficients are defined as $D(c_{i,j})$ $\forall c_{i,j} \in LL_N$. The way a set $D(c_{i,j})$ is partitioned in a sorting pass is shown in Figure 1.3. Each set $D(c_{i,j})$, such as the one shown in Figure 1.3(a), is partitioned into its four direct descendants $d_1, d_2, d_3, d_4 \in O(c_{i,j})$ as four single coefficients, and its non-direct descendants $L(c_{i,j})$ as a new subset (see Figure 1.3(b)). Later, if the $L(c_{i,j})$ subset has to be partitioned, it is divided into four subsets formed by $D(d_1)$, $D(d_2)$, $D(d_3)$ and $D(d_4)$, as shown in Figure 1.3(c). Each of these subsets can be further partitioned as we have just described. The detailed coding and decoding algorithms are described in [25]. In these algorithms, the sorting pass includes two lists to identify single coefficients: a list for the significant coefficients (called List of Significant Pixels (LSP)) and another for the insignificant ones (List of Insignificant Pixels (LIP)). On the other hand, the insignificant subsets are identified with another list (called List of Insignificant Sets (LIS)), in which each subset can be of type $D(c_{i,j})$ or $L(c_{i,j})$ (an extra tag is needed to specify it). Note that there is no list of significant subsets because when a subset is found to have a significant coefficient, it is successively partitioned until the significant coefficient or coefficients are refined to the granularity of a single coefficient.

The coding efficiency of SPIHT can be improved by using adaptive arithmetic coding to encode as a single symbol the significance values resulting from the significance tests (queries). The SPIHT algorithm has been considered a reference benchmark for wavelet image coding in a large number of papers. In addition, many papers have been published based on the tree-based SPIHT algorithm, including video coding [36, 37], hyperspectral image coding [38] and a generalization of the set partitioning algorithm [39]. Due to its similarities to EZW, the features of SPIHT are the same as those mentioned for EZW, except for the improvements in coding performance.

1.1.2.4 Lower tree wavelet encoder

Not all the tree-based algorithms in the literature are based on successive quantization implemented with bit-plane coding, leading to an embedded bit-stream. Lower Tree Wavelet (LTW) is a tree-based wavelet image encoder, with state-of-the-art coding efficiency, but less resource demanding than other encoders in the literature. The basic idea of this encoder is very simple: after computing a dyadic wavelet transform of an image, the wavelet coefficients are first quantized (using uniform scalar quantization by a factor Q) and then encoded with arithmetic coding.

In LTW [40], the quantization process is performed by two strategies: one coarser and another finer. The finer one consists in applying a scalar uniform quantization, *Q*, to wavelet coefficients. The coarser one is based on removing the least significant bit planes, *rplanes*, from wavelet coefficients. A tree structure (similar to that of [25]) is used not only to reduce data redundancy among subbands, but also as a simple and fast way of grouping coefficients. As a consequence, the total number of symbols needed to encode the image is reduced, decreasing the overall execution time. This structure is called lower tree, and it is a coefficient tree in which all its coefficients are lower than $2^{rplanes}$.

The LTW algorithm consists of two stages. In the first one, the significance map is built after quantizing the wavelet coefficients (by means of both *Q* and *rplanes* parameters). In Figure 1.6 (right) we can see the significance map built from wavelet decomposition shown in Figure 1.6 (left). The symbol set employed in this proposal is the following one: a *LOWER* symbol represents an insignificant coefficient that is the root of a lower-tree, the rest of coefficients in a lower-tree are labeled as *LOWER_COMPONENT*, but they are never encoded because they are already represented by the root coefficient. If a coefficient is insignificant but it does not belong to a lower-tree because it has at least one significant descendant, it is labeled as an *ISOLATED_LOWER* symbol. For a significant coefficient, two types of 'numeric symbols' are used according to the coefficient offspring. (a) A 'regular numeric symbol' ($nbits_{i,j}$) shows the number of bits needed to encode a coefficient, (b) and a special 'LOWER numeric symbol' ($nbits_{i,j}^{LOWER}$) not only indicates the number of bits of the coefficient, but also the fact that all its descendants are labeled as LOWER_COMPONENT, and thus they belong to a lower-tree (i.e, 4^L in Figure 1.6 (right)).

Let us describe the coding algorithm. In the first stage (symbol computation), all wavelet subbands are scanned in 2x2 blocks of coefficients, from the first decomposition level to the N^{th} (to be able to build the lower-trees from leaves to root). In the first level subband, if the four coefficients in each 2x2 block are insignificant (i.e., lower than $2^{rplanes}$), they are considered to be part of the same lower-tree, labeled as *LOWER_COMPONENT*. Then, when scanning upper level subbands, if a 2x2 block has four insignificant coefficients, and all their direct descendants are *LOWER_COMPONENT*, the coefficients in that block are labeled as *LOWER_COMPONENT*, increasing the lower-tree size. However, when at least one coefficient in the block is significant, the lower-tree cannot continue growing. In that case, a symbol for each coefficient is computed one by one. Each insignificant coefficient in the block is assigned a *LOWER* symbol if all its descendants are *LOWER_COMPONENT*, otherwise it is assigned an

ISOLATED_LOWER symbol. On the other hand, for each significant coefficient, a symbol indicating the number of bits needed to represent that coefficient is employed (see algorithm in Figure 1.4).

Function: LTWCalculateSymbols( )
Scan the first level subbands (HH_1, LH_1 and HL_1) in 2x2 blocks.
For each block B_n
if $|c_{i,j}| < 2^{rplanes}$ $\forall c_{i,j} \in B_n$
Set $c_{i,j} = LOWER\_COMPONENT$
else
For each $c_{i,j} \in B_n$)
if $|c_{i,j}| < 2^{rplanes}$
Set $c_{i,j} = LOWER$
Scan the remaining subbands (from level 2 to N) in 2x2 blocks.
For each block B_n
if $\left(|c_{i,j}| < 2^{rplanes} \wedge descendant(c_{i,j}) = LOWER\_COMPONENT\right)$
$\forall c_{i,j} \in B_n$
Set $c_{i,j} = LOWER\_COMPONENT$ $\forall c_{i,j} \in B_n$
else
For each $c_{i,j} \in B_n$)
if $|c_{i,j}| < 2^{rplanes} \wedge$ descendant($c_{i,j}$)=$LOWER\_COMPONENT$
Set $C_{i,j} = LOWER$
if $|c_{i,j}| < 2^{rplanes} \wedge$ descendant($c_{i,j}$)$\neq LOWER\_COMPONENT$
Set $c_{i,j} = ISOLATED\_LOWER$
End

Figure 1.4: Lower tree coding. Symbol computation

In order to reduce memory overhead, labels are applied by overwriting the coefficient value by an integer value associated to the corresponding label, which must be outside the possible range of significant coefficients (typically, by reusing the values in the quantized range $\left[0 \ldots 2^{rplanes}\right]$).

Finally, in the second stage (see algorithm in Figure 1.5), subbands are encoded from the LL_N subband to the first-level wavelet subbands, as shown in Figure 1.7. Observe that this is the order in which the decoder needs to know the symbols, so that lower-tree roots are decoded before its leaves. In addition, this order provides resolution scalability, because LL_N is a low-resolution scaled version of the original image, and as more subbands are being received, the low-resolution image can be doubled in size. In each subband, for each 2x2

block, the symbols computed in the first stage are entropy coded by means of an arithmetic encoder. Recall that no *LOWERCOMPONENT* is encoded. In addition, significant bits and its sign are needed for each significant coefficient and therefore binary encoded.

Function: LTWOutputCoefficients( )
  Scan subbands (from N to 1, in 2*x*2 blocks)
  For each $c_{i,j}$ in a subband
    if $c_{i,j} \neq LOWER\_COMPONENT$
      if $c_{i,j} = LOWER$
        arithmetic_output *LOWER*
      else if $c_{i,j} = ISOLATED\_LOWER$
        arithmetic_output *ISOLATED_LOWER*
      else
        $nbits_{i,j} = \lceil log_2 \left(|c_{i,j}| \right) \rceil$
        if $descendant(c_{i,j}) \neq LOWER\_COMPONENT$
          arithmetic_output $nbits_{i,j}$
        else
          arithmetic_output $nbits_{i,j}^{LOWER}$
        output $bit_{nbits_{i,j}-1} \left(|c_{i,j}| \right) \ldots bit_{rplane+1} \left(|c_{i,j}| \right)$
        output sign($c_{i,j}$)
End
Note: $bit_n(c)$ is a function that returns the n^{th} bit of c.

Figure 1.5: Lower tree coding. Output the wavelet coefficients

1.1.2.5 Space-frequency quantization

Space-Frequency Quantization (SFQ) encoder presented in [31] is a nonembedded tree-based image encoder. In order to minimize distortion for a target bit rate, this algorithm relies on: (1) the construction of trees of zero-coefficients (which is considered a space quantization) and, (2) a single common uniform scalar quantization applied to the wavelet subbands (this is the frequency quantization). The joint application of (1) and (2) is performed in an optimal manner, with the Lagrange multiplier method [41]. To this end, the algorithm tries to identify the optimal subset of coefficients to be discarded by encoding them as a quadtree, and the optimal step-size to quantize the remaining coefficients by applying a uniform scalar quantizer. In order to

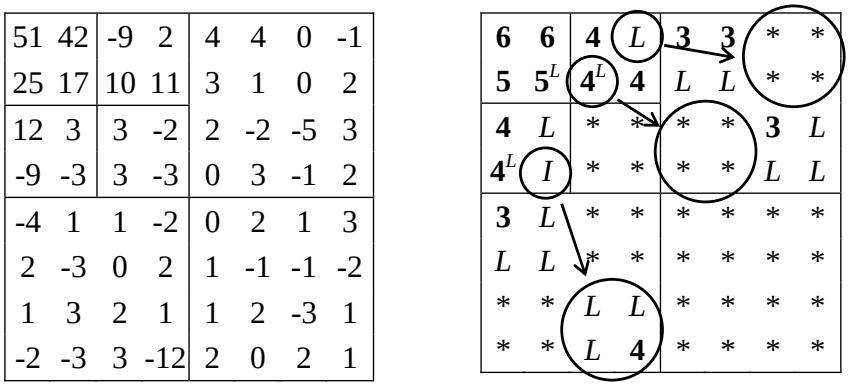


Figure 1.6: left: 2-level wavelet transform of an 8x8 example image, right: Symbol Map using *rplanes*=2

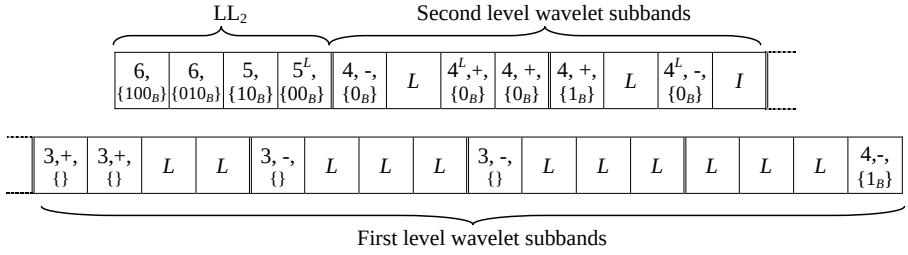


Figure 1.7: Example image encoded using LTW

determine the best option for the space quantization, the algorithm considers not only entire quad-trees, like the one shown in Figure 1.2, but also different shapes of trees, by pruning tree branches. Information about tree pruning and the rest of quantized coefficients, along with the employed step-size, are encoded with entropy coding and sent to the decoder as part of the compressed bit-stream.

Despite not being embedded, SFQ achieves precise rate control due to the use of an iterative rate/distortion optimization algorithm for a given bit rate. As a result of this algorithm, the coding performance of SFQ is slightly better than SPIHT. However, this iterative optimization algorithm is time-consuming and causes the SFQ encoder to be about five times slower than SPIHT.

1.1.2.6 Non-embedded SPIHT

In [42], Pearlman introduces the discussion about the general necessity of embedding in image coding. As we have mentioned in subsection 1.1.2.3, bit plane coding slows the execution of both the encoder and decoder, and sometimes it provides no benefit to the application, or even worse, it is not feasible. In particular, a line-based wavelet transform cannot be employed along with bit plane coding unless further rearrangement of the bit-stream is performed, needing at least the entire bit-stream in memory. On the other hand, we may just want to encode an image at a constant quality. In this case, successive approximation is not strictly required, except eventually to improve coding efficiency.

The variation of SPIHT introduced in [42] is to send all the bits down to a given bit plane (r) once a single coefficient has been found significant, so as to avoid the refinement passes. In this version, the coding process finishes when that bit plane (r) is reached in a sorting pass. Another option is to pre-quantize all the coefficients with a uniform scalar quantizer, and then encode all the bit planes (again without refinement passes). The desired distortion level (or compression level) is controlled by modifying the r parameter in the first variation, or the quantization step in the second one. Note that in both approaches, the LSP list of SPIHT is no longer needed.

Although this version is faster than the original one, neither multiple image scans nor bit plane processing of the sorting passes is avoided. Hence, the problems addressed in subsection 1.1.2.3 still remain.

1.1.2.7 Progressive resolution decomposition

The modification of SPIHT described in the previous subsection is neither SNR nor resolution scalable. Recently, the authors of SPIHT have proposed a new version of SPIHT [43] for very fast resolution scalable encoding, based on the principles of decreasing energy of the wavelet coefficients along the subband levels, and the fact that the energy is quite similar for coefficients at the same level. Since it supports resolution scalability with great speed, the authors consider that it is an excellent choice for remote sensing and Geographic Information System (GIS) applications, where rapid browsing of various scales of large images is necessary.

Progressive resolution decomposition (PROGRESS) uses a pre-defined constant quality factor, just like the non-embedded SPIHT algorithm. In order to reduce complexity, bit plane coding is avoided and each coefficient is visited only once. Entropy coding is also avoided.

For each coefficient, the goal is to encode the sign and the bits below the most significant non-zero bit. To this end, the number of bits required for each coefficient must be known in advance. Basically, at a subband level, for each coefficient $c_{i,j}$ in that subband, the PROGRESS algorithm identifies the number of bits needed to encode the highest coefficient in a SPIHT-like subset $D(c_{i,j})$ (let us call this value r), and then it encodes each coefficient contained in $O(c_{i,j})$ with that number of bits. In order that the decoder can reconstruct the original coefficients, r is also encoded. In the next subband level, PROGRESS repeats the same operation for each $D(d_{m,n})\forall d_{m,n} \in O(c_{i,j})$. This algorithm is repeated through the successive subband levels, from the LL_N subband down to the first subband level. However, when the number of bits needed to encode a subset is found to be zero, a group of insignificant coefficients has been identified, and then this subset is no longer partitioned and encoded.

In order to improve coding efficiency, each r for a given subset is not encoded as a single value, but rather as the difference between that value in this subset and in its parent subset (i.e., the direct subset from which a subset stems). Since this difference is always positive (or zero), and its probability distribution is higher as it approaches zero, unary coding[1] is employed. Some other implementation details and the complete encoding algorithm are given in [43].

Experimental results show that PROGRESS is up to two times faster in coding and four times faster in decoding than the binary version of SPIHT (i.e., SPIHT without entropy coding). However, its coding efficiency is relatively poor, being slightly worse than binary SPIHT. The low coding performance is not only due to its lack of entropy coding, but also because it always employs the number of bits required by the highest coefficient in a subset. This problem especially affects highly detailed images. These images are more likely to have high descendant coefficients, which could cause their parents to use more bits than actually needed.

1.1.3 Image coding standard: JPEG 2000

1.1.3.0.1 Embedded Block Coding with Optimized Truncation (EBCOT)

The EBCOT [28] encoder is certainly the most important block-based wavelet encoder reported in the literature. This encoder is a refined version of the Layered Zero Coding (LZC) technique proposed by Taubman and Zakhor in [44]. The importance of EBCOT lies in the fact that it was selected to be included as the coding subsystem of the JPEG 2000 standard [45]. EBCOT

[1] In unary coding, a number n is represented with n ones followed by a zero.

achieves most requirements of JPEG 2000, such as a rich embedded bit-stream with advanced scalability, random access, robustness, etc., by means of block-based coding for the reasons given above. Furthermore, the decrease in coding efficiency caused by the lack of inter-band redundancy removal is compensated by the use of more contexts in the arithmetic encoder, a finer-granularity coding algorithm (with three passes per bit plane instead of two), and a Post-Compression Rate Distortion (PCRD) optimization algorithm based on the Lagrange multiplier method.

Due to the importance of EBCOT in the JPEG 2000 standard, we will describe it in some detail. For a more complete and general description, there are many other references such as [46, 47, 48] or even the standard document [45]. Note that the EBCOT algorithm originally published by Taubman in [28] was slightly changed for the JPEG 2000 standard in order to reduce complexity and other issues. We will focus on this adapted version.

After applying the DWT to the image, the EBCOT algorithm encodes the wavelet coefficients in fixed-size code blocks. In this first step, called tier 1 coding, each code block is completely and independently encoded, getting in this manner an independent bit-stream for each code block. Then, in the second step, tier 2 coding, fragments of bit-stream of each codeblock are selected to achieve the desired target bit rate (rate control) in an optimal way (i.e., minimizing distortion), and it is arranged in such a way so that the selected scalability is accomplished.

Prior to EBCOT, a uniform scalar quantization with deadzone is applied to the wavelet coefficients. All the code blocks in the same subband are quantized with the same step-size so that blocking artifacts are avoided. Therefore, in general, this quantization has little rate control meaning, which is performed later in tier 2 coding. Rather, it is used to balance the importance of the coefficient values (recall that the DWT employed in JPEG 2000 avoids dynamic range expansion but is not energy preserving), and in a practical way, to convert the floating point coefficients resulting from most wavelet transforms into integer data. Another way to select the quantizer step size is depending on the perceptual importance of each subband to improve visual quality based on the human visual system [49, 50, 51].

Regarding the code block size, the total number of coefficients in a block should not exceed 4096, and both width and height must be an integer power of two. Thereby, the typical code block size is 64x64, although other smaller sizes can be used (e.g., for memory saving or complexity issues). Of course, once a block size is determined, smaller code blocks can appear on the subband boundary or in subbands smaller than a regular block.

Block coding: tier 1 coding. Once the wavelet subbands are divided into blocks, an independent bit-stream is generated from each code block in the tier 1 coding stage. Each bit-stream is created with a special adaptive binary arithmetic encoder with several contexts called MQ-coder [52]. The MQ-coder is a reduced-complexity version of the usual arithmetic encoder [12], limited to coding binary symbols. The JPEG 2000 standard document [45] gives a detailed flowchart description of this encoder.

In this stage, each code block is encoded bit plane by bit plane, starting from the most significant non-zero bit plane. For each bit plane, several passes are given in order to identify the coefficients which become significant in this bit plane, and to encode the significant bits of those coefficients found significant in previous bit planes. This working philosophy is shared by many other well known encoders like EZW and SPIHT. However, unlike these encoders, three passes (instead of two) are given for each bit plane[2]. In the first pass, called significance propagation pass, the significance of the coefficients that were insignificant in previous bit planes but are likely to become significant in this bit plane are encoded. Then, in the second pass, called magnitude refinement pass, a refinement bit is encoded for each coefficient found significant in a previous bit plane. Finally, the significance of the rest of coefficients (i.e., those that were insignificant and are likely to remain insignificant in this bit plane) are encoded in the third pass, called clean-up pass.

In tier 2 coding, the bit-stream resulting from several contiguous full passes are selected from each code block to build the final bit-stream. Therefore, the bit-stream generated from each pass is the lowest granularity for the final bit-stream formation. In each code block, the point in which its bit-stream is truncated to contribute to the final bit-stream for a given bit rate is called the optimal truncation point. Figure 1.8 illustrates the encoding process and gives an example of truncation points.

The order of the passes has been decided according to their contribution to rate/distortion improvements, so that a pass that is more likely to introduce more reduction of distortion with a lower rate increase is encoded in first place. Of course, after encoding the three passes, the same reduction of distortion and the same bit rate is reached independently of the order of the passes. However, the proposed order yields more benefits if the truncation point is not at the end of a bit-plane coding (i.e., it is not between a clean-up pass and a significance propagation pass), but in the middle of it.

If we compare this algorithm with EZW or SPIHT in broad terms, we see

[2]The original EBCOT algorithm [28] had four passes instead of three.

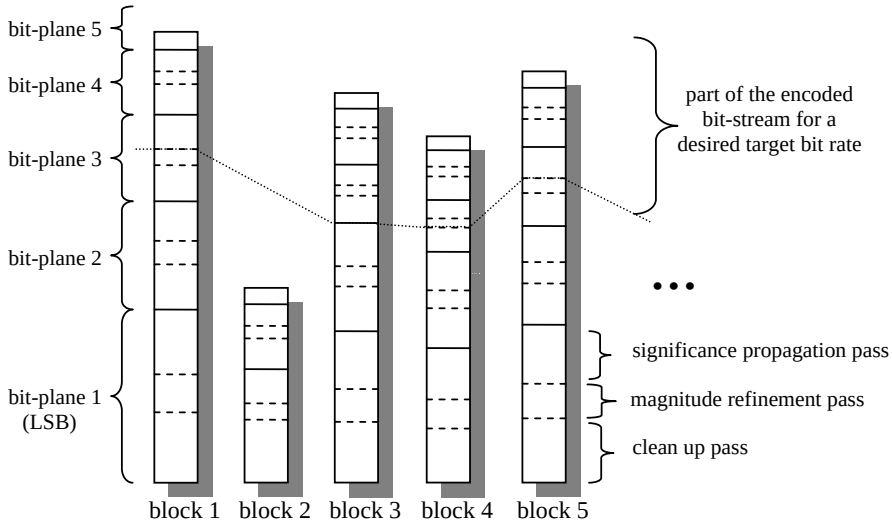


Figure 1.8: Example of block coding in JPEG 2000. In tier 1 coding, each code block is completely encoded bit plane by bit plane, with three passes per bit plane (namely signification propagation, magnitude refinement and clean-up passes). Only part of each code block is included in the final bit-stream. In this figure, the truncation point for each code block is pointed out with a dotted line. These truncation points are computed with an optimization algorithm in tier 2 coding, in order to match the desired bit rate with the lowest distortion

that the main difference (apart from the lack of trees) is that the pass employed to identify new significant coefficients (called dominant pass in EZW and sorting pass in SPIHT) has been split into two passes in order to have more passes from which to choose a truncation point.

For implementation convenience, the order in which coefficients are scanned in a codeblock is in stripes formed by columns of four coefficients, as shown in Figure 1.9(a).

Let us see more details of each coding pass. In the significance propagation pass, a coefficient is said to be likely to become significant if, at the beginning of that pass, it has at least one significant neighbor. Certainly, this condition does not guarantee that it will become significant in this bit plane, and therefore its significance still has to be encoded. In order to improve coding efficiency, nine contexts are used according to the significance of its eight immediate neighbors (see Figure 1.9(b)). The exact context assignment, mapping from the $2^8 - 1$ possible contexts to nine contexts, can be found in [28]. In addition, when a coefficient eventually becomes

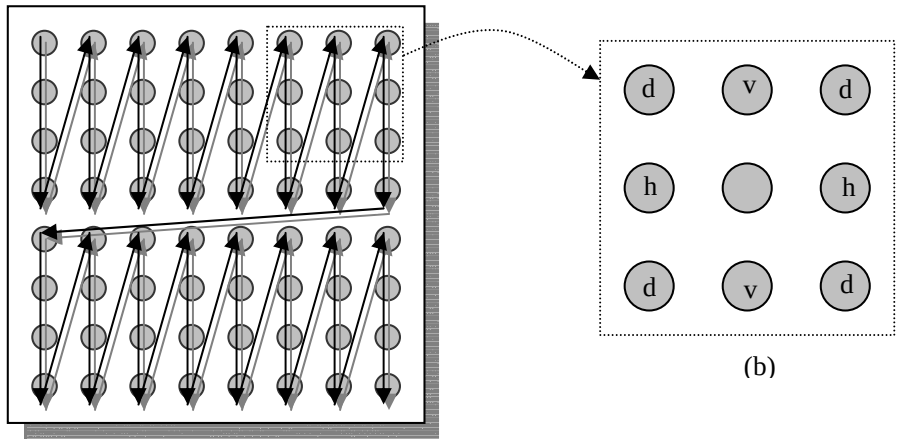


Figure 1.9: (a) Scan order within an 8x8 code block in JPEG 2000, and (b) context employed for a coefficient, formed by its eight neighbor coefficients (two horizontal, two vertical, and four diagonal)

significant, its sign is also arithmetically encoded with five different contexts.

In the case of the magnitude refinement pass, a refinement bit is arithmetically encoded with two contexts if it has just become significant in the previous bit plane (i.e., it is the first bit encoded for this coefficient). For the rest of bits, they are considered to have even distribution and thereby another single context is used without dependence of the neighboring values.

The clean-up pass is implemented in a similar manner to the signification propagation pass, with the same nine contexts employed to encode the significance of a single coefficient. However, the clean-up pass includes a novel run mode, which serves to reduce complexity, rather than improve coding efficiency. Observe that most coefficients are insignificant in this pass, and therefore the same binary symbol is encoded many times. We can reduce complexity if we take advantage of this fact and reduce the number of encoded symbols. To this end, when four coefficients forming a column have insignificant neighbors, a run mode is entered. In this mode, we do not encode single coefficients but a binary symbol that specifies if any of the four coefficients in a column is significant. This binary symbol is encoded with a single context.

Note that, for the most significant non-zero bit plane (i.e., the first bit plane that is encoded), neither a significance propagation pass nor a magnitude refinement pass is performed, because there is no previous significant

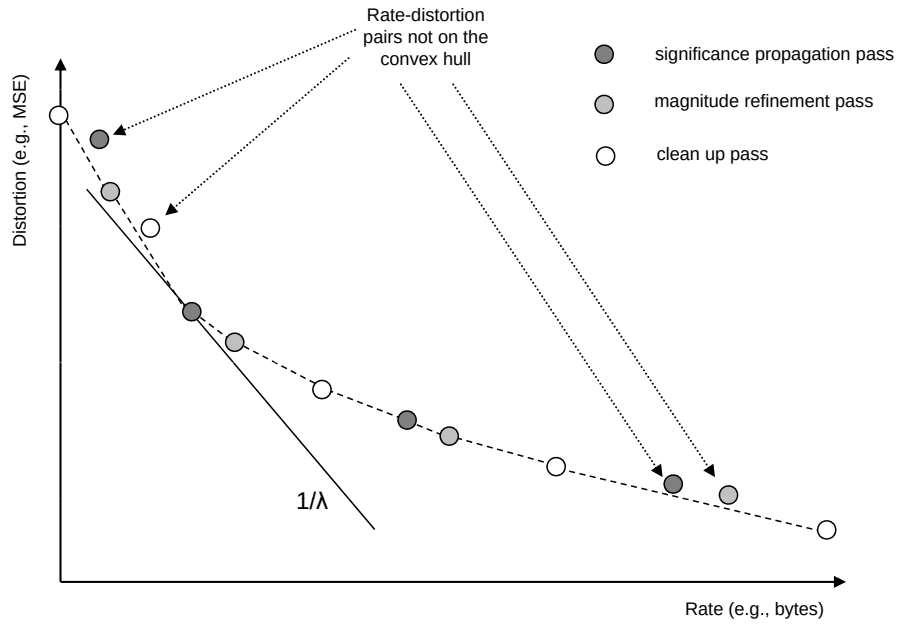


Figure 1.10: Example of convex hull formed by distortion-rate pairs from block 1 of Figure 1.8. In a convex hull, the slopes must be strictly decreasing. Four rate-distortion pairs are not on the convex hull, and therefore they are not eligible for the set of possible truncation points. A line with a slope of $1 \div \lambda$ determines the optimal truncation point for a given value of λ

coefficient (see example in Figure 1.8). Finally, it is also worth mentioning that, from the above description, we can deduce that the MQ-coder must be able to support (at least) eighteen contexts.

Bit-stream organization: tier 2 coding. In tier 2 coding, the bit-streams generated from each code block are multiplexed using a specific file format to accomplish the desired scalability. Rate control tasks are also performed in this second stage.

In order to determine the optimal truncation point in each code block for a desired bit rate, EBCOT proposes a post-compression rate distortion (PCRD) optimization algorithm, which is basically a variation of the Lagrange multiplier method [41]. This algorithm computes a convex hull (where slopes must be strictly decreasing) for each code block from a set of distortion-rate pairs (see Figure 1.10 for an example of a convex hull). Each pair defines the contribution of a coding pass to reduce image distortion (e.g., measured as

Mean Squared Error (MSE) reduction) and the cost of that pass (e.g., the number of bytes required to encode that pass). For an optimal bit-stream formation, the rate-distortion pairs in the interior of the convex hull cannot be selected as truncation points.

Given the set of convex hulls for each code block, an optimal bit-stream can be achieved as follows. Consider a factor λ that defines a straight line with $1 \div \lambda$ slope. The optimal truncation point for each convex hull is given by the point to which that line is "tangent-like"[3]. In other words, it is the point at which the rate/distortion slope changes from being greater than $1 \div \lambda$ to less than it (see example in Figure 1.10). In this way, we can compute an optimal bit-stream by calculating a truncation point for each code block with a given λ. However, no rate control is performed. In order to achieve a target bit rate, the value of λ is iteratively changed and the optimal set of truncation points are recomputed with each value of λ. From all the sets of truncation points iteratively computed that do not exceed the desired bit rate, the one that yields the highest distortion reduction is selected. In other words, we select the largest bit-stream that does not exceed the target bit rate.

Quality (SNR) scalability can be achieved if this rate control algorithm is executed several times, once for each partial target bit rate (R_1, R_2,..., R_n). Therefore, the selected coding passes that optimally lead to a bit rate R_1 are said to form the quality layer 1; then, the added coding passes that lead to a bit rate R_2 form the quality layer 2, and so on. In this way, EBCOT produces an embedded bit-stream, but with a coarser granularity than the one of EZW and SPIHT. On the other hand, for resolution scalability, we just have to arrange the selected code blocks depending on the subband level, from the LL_N to the first-level wavelet subbands. A wide variety of types of scalability is accomplished by combining various quality layers and the suitable code block arrangement in the final bit-stream.

Performance and complexity analysis. Although EBCOT only exploits intra-block redundancy, it generally performs as well as SPIHT, or even better than it, in terms of coding efficiency, mainly due to (1) the use of more contexts, (2) the introduction of a third pass to encode the most important information in first place, and (3) the PCRD optimization algorithm. In addition, if we consider artificial images or highly detailed natural images, EBCOT clearly outperforms SPIHT, because in this type of image, SPIHT can establish fewer

[3]Formally speaking, the given convex hulls are not curves and then we cannot consider a line tangent to it. Here, we actually mean a line that touches a convex hull and does not intersect it. Note that in the case of a curve, there is only a line tangent to each point, whereas in our convex hulls, there are many "tangent-like" lines for each possible truncation point.

coefficient trees, and also due to the use of more contexts in EBCOT, enabling a better and more precise adaptation of its probability model.

Let us perform a complexity analysis of EBCOT. Recall that the main complexity problem in SPIHT is introduced by bit-plane coding. Nonetheless, although both EBCOT and SPIHT use bit-plane coding, EBCOT avoids the locality problems that increase the cache miss rate by encoding an image block-by-block. Moreover, the set of code block bit-streams is more likely to fit into the cache, and therefore further post-processing does not cause so many cache misses. In spite of this, the EBCOT algorithm can be considered more complex than SPIHT (except for very large images in cache-based systems). There are several reasons for this. First, bit plane coding is still present, and for each bit plane, it must be performed for all the coefficients in a block. Compare it with EZW and SPIHT, where the coefficients in a tree are neither encoded nor scanned. Second, the significance analysis is more complex in EBCOT, since more contexts are used. Third, in a regular implementation of EBCOT, each coefficient is fully encoded, bit plane by bit plane, despite the fact that some bit planes will not be included in the final bit-stream due to rate control restrictions although some advanced implementations of JPEG 2000 perform a conservative heuristic for incrementally estimating the number of coding passes that will be included in the final bit-stream, and determine those bit planes that do not need to be computed. Finally, the PCRD optimization algorithm is performed and it is an iterative process.

1.2 Video coding

Television won't last. It's a flash in the pan.
(Mary Somerville, radio presenter, in 1948)

1.2.1 Fundamentals

Compression is an almost mandatory step in storage and transmission of video, since, as simple computation can show, one hour of color video at International Radio Consultative Committee - Comité Consultatif International des Radiocommunications (CCIR) 601 resolution (576x704 pixels per frame) requires about 110 Giga Byte (GB) for storing or 240 Mega bits per second (Mbps) for real time transmission.

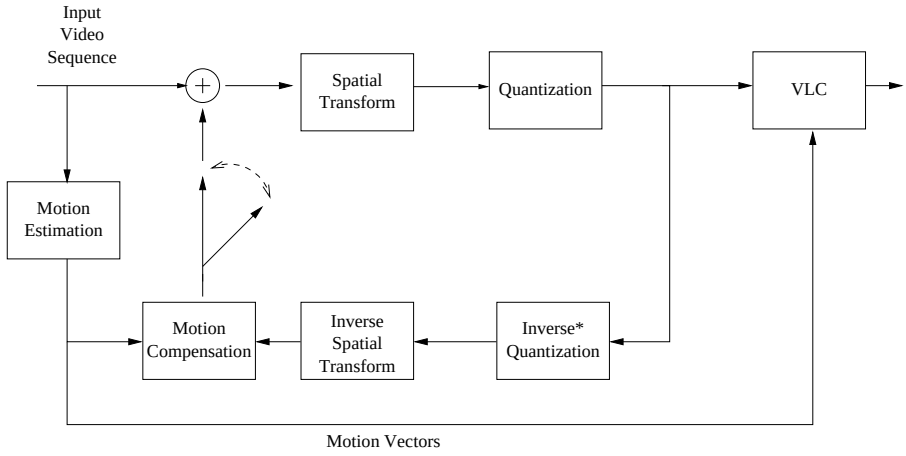


Figure 1.11: General Scheme of a Hybrid Video Encoder

On the other hand, video is a highly redundant signal, as it is made up of still images (called frames) which are usually very similar to one another, and moreover are composed by homogeneous regions. The similarity among different frames is also known as temporal redundancy, while the homogeneity of single frames is called spatial redundancy. Most video encoders perform their job by exploiting both kinds of redundancy and thus using a spatial analysis (or spatial compression) stage and a temporal analysis (or temporal compression) stage.

1.2.1.1 Hybrid video coding

The most successful video compression schemes to date are those based on Hybrid video coding. This definition refers to two different techniques used in order to exploit spatial redundancy and temporal redundancy. Spatial compression is indeed obtained by means of a transform based approach, which makes use of the DCT, or its variations. Temporal compression is achieved by computing a Motion-Compensated (MC-ed) prediction of the current frame and then encoding the corresponding prediction error. Of course, such an encoding scheme needs a Motion Estimation (ME) stage in order to find Motion information necessary for prediction.

A general scheme of a hybrid encoder is given in Figure 1.11. Its main characteristics are briefly recalled here.

The hybrid encoder works in two possible modes: Intraframe and Interframe. In the intraframe mode, the current frame is encoded without any reference to other frames, so it can be decoded independently from the others. Intra-coded frames (also called anchor frames) have worse compression performances than inter-coded frames, as the latter benefits from Motion-compensated prediction. Nevertheless they are very important as they assure random access, error propagation control and fast-forward decoding capabilities. The intra frames are usually encoded with a JPEG-like algorithm, as they undergo DCT, Quantization and Variable Length Coding (VLC). The spatial transform stage concentrates signal energy in a few significative coefficients, which can be quantized differently according to their visual importance. The quantization step here is usually tuned in order to match the output bit rate to the channel characteristics.

In the interframe mode, the current frame is predicted by Motion compensation from previously encoded frames. Usually, Motion-Compensated prediction of the current frame is generated by composing blocks taken at displaced positions in the reference frame(s). The position at which blocks should be considered is obtained by adding to the current position a displacement vector, also known as Motion Vector (MV). Once current frame prediction is obtained, the prediction error is computed, and it is encoded with the same scheme as intra frames, that is, it undergoes a spatial transform, quantization and entropy coding.

In order to obtain Motion vectors, a ME stage is needed. This stage has to find which vector better describe current block motion with respect to one (or several) reference frame. Motion Vectors have to be encoded and transmitted as well. A VLC stage is used at this end.

All existing video coding standards share this basic structure, except for some MPEG-4 features. The simple scheme described so far does not integrate any scalability support. A scalable compressed bit-stream can be defined as one made up of multiple embedded subsets, each of them representing the original video sequence at a particular resolution, frame rate or quality. Moreover, each subset should be an efficient compression of the data it represents. Scalability is a very important feature in network delivery of multimedia (and of video in particular), as it allows encoding the video just once, while it can be decoded at different rates and quality parameters, according to the requirements of different users.

The importance of scalability was gradually recognized in video coding standards. The earliest algorithms (as ITU H.261 norm [53, 54]) did not provide scalability features, but as soon as MPEG-1 was released [55], the standardization boards had already begun to address this issue. In fact, MPEG-1 scalability is very limited (it allows a sort of temporal scalability thanks to the subdivision in Group of Pictures (GOP). The following International Organization for Standardization (ISO) standards, MPEG-2 and MPEG-4 [56, 57, 58] increasingly recognized scalability importance, allowing more sophisticated features. MPEG-2 compressed bit-stream can be separated in subsets corresponding to multiple spatial resolutions and quantization precisions. This is achieved by introducing multiple motion compensation loops, which, on the other hand, involves a remarkable reduction in compression efficiency. For this reason, it is not convenient to use more than two or three scales.

Scalability issues were even more deeply addressed in MPEG-4, whose Fine Grain Scalability (FGS) allows a large number of scales. It is possible to avoid further Motion Compensation (MC) loops, but this comes at the cost of a drift phenomenon in motion compensation at different scales. In any case, introducing scalability affects significantly performances. The fundamental reason is the predictive MC loop, which is based on the assumption that at any moment the decoder is completely aware of all information already encoded. This means that for each embedded subset to be consistently decodable, multiple motion compensation loops must be employed, and they inherently degrade performances. An alternative approach (always within a hybrid scheme) could provide the possibility, for the local decoding loop at the encoder side, to lose synchronization with the decoder at certain scales; otherwise, the enhancement information at certain scales should ignore motion redundancy. However, both solutions degrade performances at those scales.

The conclusion is that hybrid schemes, characterized with a feedback loop at the encoder, are inherently limited in scalability.

1.2.2 Video coding standard: H.264

The encoder (shown in Figure 1.12) has two paths known as the *forward path* (left to right) and the *reconstruction path* (right to left). In the *forward path* an input frame or field F_n is processed in MBs (16x16 pixels), and can be coded in *Intra* or in *Inter* mode. The encoder creates a reconstructed frame (*P*), based on reconstructed pictures samples. In *Intra* mode, *P* is formed from samples in the current slice that have been previously encoded, decoded and reconstructed (uF_n in the Figure 1.12). In the Inter mode, *P* is created by *Motion Compensation* (MC) prediction from the reference pictures. These reference pictures may be chosen from a selection of past or future pictures that have already been encoded, reconstructed and filtered. This prediction image (*P*) is subtracted from the current image to produce a residual image, which will be transformed and quantized to obtain *X*, a set of quantized transform coefficients which are reordered and entropy encoded. The encoder also decodes the frame to provide a reference for future predictions. The *X* image is scaled (Q^{-1}) and inverse transformed (T^{-1}) to produce *Dn*. The *P* image is added to D_n to create the reconstructed image uF_n. However, this image is unfiltered. In the last step, a filter is used to reduce the effects of blocking distortion.

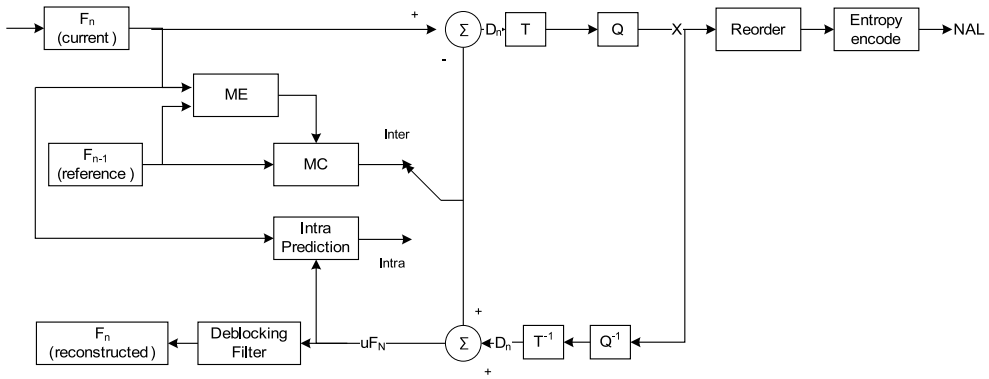


Figure 1.12: Block Diagram for an H.264 encoder

A Deblocking Filter is used to reduce blocking distortion and is applied to each decoded macroblock. This module may improve the compression performance, because the filtered image is often a more reliable reproduction of the original frame than a block and unfiltered image. In the encoder (see Figure 1.12) this filter processes the macroblock after the inverse transform T^{-1}, prior to the stage of reconstruction and storing for future predictions. In the decoder (Figure 1.13), it is the last operation of the process. The function

of this module is to smooth block edges, improving the appearance of the decoded frames. The filtered image is used for motion compensation in future frames. The filter is applied to vertical and horizontal edges of 4x4 blocks in a macroblock but the edges on slices boundaries.

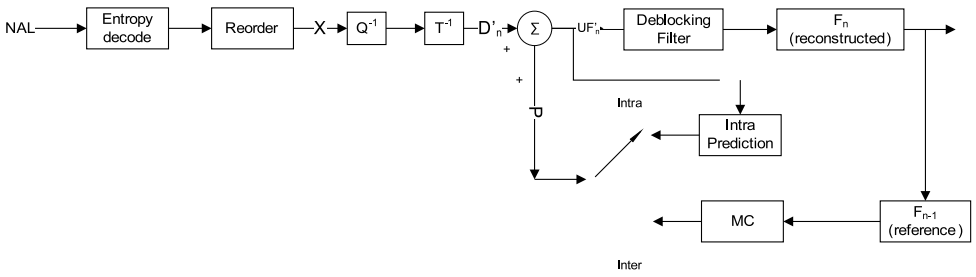


Figure 1.13: Block Diagram for an H.264 decoder

The Transform, used in the H.264 standard, T and T^{-1}, depends on the type of residual data to be coded. There are three kinds of transforms available: a *Hadamard Transform* (HT) for the 4x4 array of luminance *Dominant Component* (DC) coefficients in Intra MBs predicted in 16x16 mode, a HT for the 2x2 array of chrominance DC coefficients in any macroblock and a DCT-based transform for all other 4x4 blocks in the residual data. The H.264 transform [59] is based on the DCT but with some fundamental differences:

- It is an integer transform, which implies no floating point operations are needed. The mismatch between the encoder and the decoder is zero without loss of accuracy.
- It can be implemented using only additions and shifts.
- The number of operations can be reduced by integrating part of the operations involved in the transform into the quantizer.

As depicted in the H.264 reference standard [60] the two dimensional DCT transform is implemented applying a one-dimensional DCT transform twice, one to the horizontal dimension and another to the vertical one [61]. In the first step, the horizontal correlation within the nxn samples block is exploited and in the second step the one-dimensional DCT transform is applied to exploit the vertical correlation.

The transformation matrix H is a 4x4 matrix defined as [60] in 1.1:

$$H = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{pmatrix} \tag{1.1}$$

The inverse transformation matrix H_{inv} is a 4x4 matrix defined as [60] in 1.2:

$$H_{inv} = \begin{pmatrix} 1 & 1 & 1 & \frac{1}{2} \\ 2 & \frac{1}{2} & -1 & -1 \\ 1 & -\frac{1}{2} & -1 & 1 \\ 1 & -1 & 1 & -\frac{1}{2} \end{pmatrix} \tag{1.2}$$

The relationship between the matrices H_{inv} and H is given by equation 1.3), where I is the Identity matrix:

$$H_{inv} \begin{pmatrix} \frac{1}{4} & 0 & 0 & 0 \\ 0 & \frac{1}{5} & 0 & 0 \\ 0 & 0 & \frac{1}{4} & 0 \\ 0 & 0 & 0 & \frac{1}{5} \end{pmatrix} H = I \tag{1.3}$$

The Quantizer, Q and Q^{-1} (Figure 1.12 and Figure 1.13), adopted by the H.264 standard, is a scalar quantizer. A total of 52 values for the *Quantification Parameter* (QP) are supported by the standard. The quantification step is doubled in size for every increment of 6 in QP. The wide range of quantizer step sizes makes it possible for an encoder to control the trade-off accurately and flexibly between bit rate and quality. Besides, the H.264 standard allows different values for the QP for luminance and chrominance. The quantization step-sizes are not linearly related to the quantization parameter (as in all prior standards). A default relationship is specified between the quantization step sizes used for luminance and chrominance, and the encoder can adjust this relationship at the slice level to balance the desired fidelity of the color components.

The Entropy encoding (Figure 1.12) or the Entropy decoding (Figure 1.13) are the modules where the elements of the sequence are encoded/decoded, using fixed or variable length binary codes. As shown later, this operation depends of the profile being used to encode/decode the video sequence.

The entropy-encoded coefficients, together with side information required to decode each macroblock from the compressed bit-stream pass to the *Network Abstraction Layer* (NAL) where the picture will be prepared for

transmission or storage. The H.264 standard does not specify the mechanism of transmitting NAL units, but a distinction is made between transmission over packet-based transport mechanisms (packet networks) and transmission in a continuous data stream (circuit-switched channels). Each NAL unit contains a *Raw Byte Sequence Payload* (Raw Byte Sequence Payload (RBSP)), a set of data corresponding to coded video data or header information. The reason to use variable code lengths and NAL is to discriminate between coding and transport features.

On the other hand, the decoder (Figure 1.13) only has the forward path (left to right). The data flow path in the decoder shows the similarities between encoder and decoder.

The input for the decoder is a compressed bit-stream from the NAL, and the entropy module decodes the data to generate a set of quantized coefficients, denoted by X in Figure 1.13. These are scaled and inverse transformed to give $D'n$, exactly the same $D'n$ created in the encoder (Figure 1.12), in case that there were no errors during the process. Using the information stored in the video sequence, the decoder generates the P image. The decoder adds these two images to produce $uF'n$, which will be filtered to obtain $F'n$.

In the decoder (Figure 1.13), each block coming from the quantizer is mapped into a sixteen element array in a zig-zag order. This is the function made by the reorder. This module has the function to prepare the data (reordering the coefficients for optimization) for the next module, where the entropy coding is performed. The inverse process is made by the decoder (Figure 1.13). The MB coefficients are reordered before the inverse quantification.

1.2.2.1 H.264 Inter prediction

There are some concepts redefined in the H.264 standard which will be used in the next sections. They are summarized in the following paragraphs:

The *fields* and the *frames* are used in a different way. Both can be encoded to produce a coded picture of interlaced video, however only a frame can be coded using progressive video. The decoding order is not necessarily related to the number of frames of each encoded frame. Each coded field or frame has an associated picture order count, which defines the decoding order. Previously coded pictures, reference pictures, may be used for Inter prediction of further coded pictures.

A coded picture consists of a number of MBs, each containing 16x16 luminance samples and associated 8x8 chrominance samples (Chrominance

blue (Cb) and Chrominance red (Cr) in the H.264 standard) if any, depending on the sampling format. Within each picture, MBs are ordered in slices, where a slice is a set of MBs in raster scan order, but not necessarily contiguous. An I slice may contain only I MBs types, a P slice may contain P and I MB types and a B slice may contain B and I MB types.

The MB prediction (Intra or Inter) is performed in the H.264 standard using previously encoded data. In the case of the Intra prediction, an Intra MB is predicted from the current slice after having been encoded, decoded and reconstructed. For the Inter prediction, the MB is predicted using samples previously encoded. The MB prediction and the current MB are subtracted, and the result is compressed and transmitted to the decoder, together with the information required for the decoder to repeat the prediction process (motion vectors, prediction mode, etc.). The decoder needs this information to create the prediction and adds the residual to it. The encoder must encode and decode the sequence to make sure that the decoder will have the same reconstructed information.

H.264 allows 4:2:0 progressive or interlaced video. In the default sampling format (4:2:0), chrominance samples (Cb and Cr) are aligned horizontally with every 2nd luminance sample and are located vertically between two luminance samples. Chrominance components have half the horizontal and vertical resolution of the luminance component.

The basic mechanism used to encode the residual is the *Context Based Adaptive Variable Length Coding* (Context Based Adaptive Variable Length Coding (CAVLC)) [62]. CAVLC uses run-level coding to represent strings of zeros compactly. The number of coefficients is encoded using a look-up table, and the choice depends on the number of nonzero coefficients in neighboring blocks. This mechanism can take advantage, just in case the magnitude of nonzero coefficients tends to be larger at the start of the reordered array, and smaller towards the higher frequencies. CAVLC chooses the entry of *Variable Length Code* (VLC) look-up table for the level parameter, depending on recently coded level magnitudes.

In the H.264 standard, the MB mode decision in Inter frames (those where the motion estimation is carried out) is the most computationally expensive process due to the use of the variable block-size, motion estimation, quarter-pixel motion compensation, etc. *Inter prediction* creates a prediction model from one or more previously encoded video frames or fields using *block based motion compensation* as depicted in Figure 1.14.

H.264 uses *block-based motion compensation*, the same principle adopted by every major coding standard since H.261. Important differences from

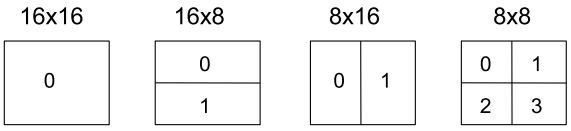


Figure 1.14: MB partitions: 16x16, 16x8, 8x16 and 8x8

earlier standards include the support for a range of block sizes (down to 4x4) and fine sub-pixel motion vectors (1/4 pixel in the luminance component). H.264 supports motion compensation block sizes ranging from 16x16 to 4x4 luminance samples with many options between the two.

The luminance component of each MB (16x16 samples) may be divided into four different ways (Figure 1.14): one 16x16 MB partition, two 16x8 partitions, two 8x16 partitions or four 8x8 partitions. Each of the sub-divided regions is a MB partition. If the 8x8 mode is chosen, each of the four 8x8 MB partitions within the MB may be further separated into four different ways (Figure 1.15): one 8x8 partition, two 8x4 partitions, two 4x8 partitions or four 4x4 partitions (known as *sub-macroblock partitions*). These partitions and sub-partitions give rise to a large number of possible combinations within each macroblock. This method of partitioning MBs into motion compensated sub-blocks of varying size is known as *tree structured motion compensation*.

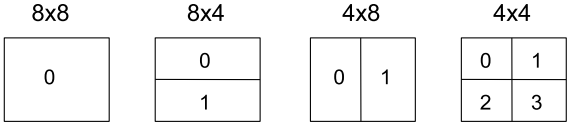
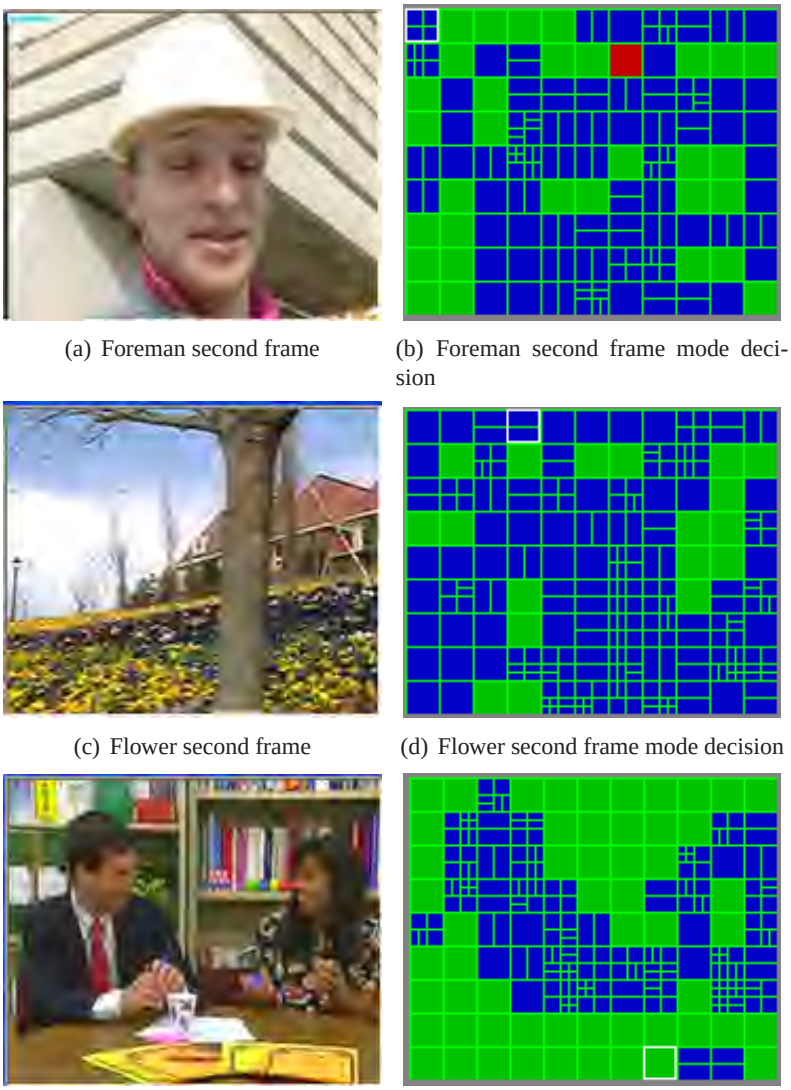


Figure 1.15: Sub-macroblock partitions: 8x8, 8x4, 4x8 and 4x4

The resolution of each chrominance component in a macroblock (Cr and Cb) is half that of the luminance component. Each chrominance block is partitioned in the same way as the luminance component, except that the partition sizes have exactly half the horizontal and vertical resolution (an 8x16 partition in luminance corresponds to a 4x8 partition in chrominance; an 8x4 partition in luminance corresponds to 4x2 in chrominance and so on). The horizontal and vertical components of each motion vector (one per partition) are halved when applied to the chrominance blocks.

Figure 1.16 shows the second frame of sequences *Foreman*, *Flower and Garden* and *Paris*, and their mode decisions made by the *Inter prediction*, in the *Baseline Profile* with all parameters as default. In the example, the best



(a) Foreman second frame

(b) Foreman second frame mode decision

(c) Flower second frame

(d) Flower second frame mode decision

(e) Paris second frame

(f) Paris second frame mode decision

Figure 1.16: Inter prediction in H.264

match for the present current block is given for the mode that has the smallest *Sum Absolute Differences* (SAE). See 1.17 for legend.

In order to evaluate the *motion vectors*, each partition in an inter-coded MB is predicted from an area of the same size in a reference picture. The offset between the two areas (the motion vector) has 1/4-pixel resolution (for the luminance component). If the video source sampling is 4:2:0, 1/8 pixel samples are required in the chrominance components (corresponding to

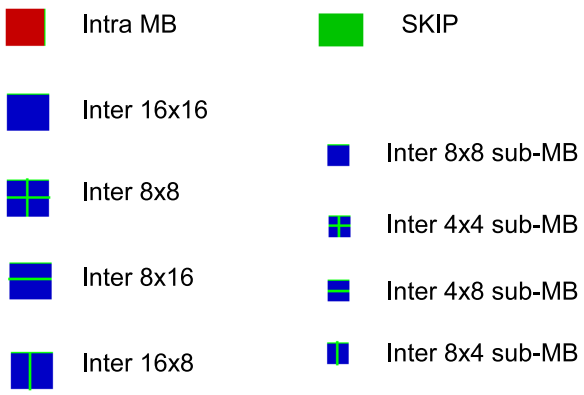


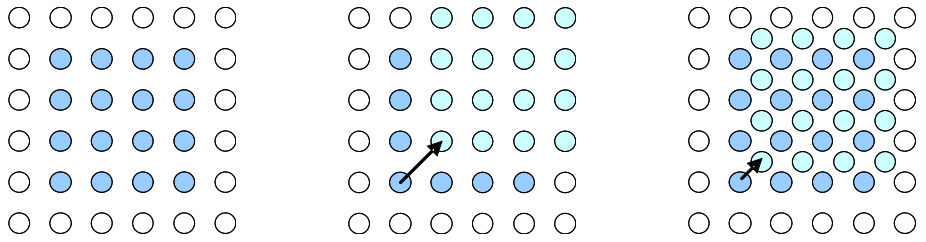
Figure 1.17: Different kinds of Inter MBs in Figure 1.16

1/4-pixel samples in the luminance). The luminance and chrominance samples at sub-pixel positions do not exist in the reference picture and so it is necessary to create them using interpolation from nearby image samples. For example, in Figure 1.18, a 4x4 block in a frame is predicted from a region of the reference picture in the neighborhood of the current position. If the horizontal and vertical components of the motion vectors are integers, the relevant samples in the reference block actually exist. If one or both vectors components are fractional values, the prediction samples are generated by interpolation between adjacent samples in the reference frame. Sub-pixel motion compensation can provide significantly better compression performance than integer-pixel compensation, at the expense of increased complexity. Quarter-pixel accuracy outperforms half-pixel accuracy.

Encoding a motion vector for each partition can take a significant number of bits, especially if small partition sizes are chosen. Motion vectors for neighboring partitions are often highly correlated and therefore each motion vector is predicted from vectors of nearby, previously coded partitions. The method of forming a predicted motion vector depends on the motion compensation partition size and on the availability of nearby vectors.

1.2.2.2 H.264 Intra prediction

H.264 incorporates an Intra picture prediction into its coding process (defined within the pixel domain) whose main aim is to improve the compression efficiency of the Intra coded pictures and Intra MBs. Intra prediction can result



(a) 4x4 block current frame (b) Reference vector (1,-1) (c) Reference vector (0.75,-0.5)

Figure 1.18: 4x4 example of integer and sub-sample prediction

in significant savings when the motion present in the video sequence is minimal and the spatial correlations are significant. Throughout this section, the principle of operation of the Intra frame prediction modes as applied to the luminance and chrominance blocks will be illustrated.

While macro blocks of 16x16 pixels are still used, predicting an MB from the previously encoded MBs in the same picture is new in H.264. For luminance component, an MB may make use of 4x4 and 16x16 block prediction modes, referred to as Intra_4x4 and Intra_16x16, respectively. Recently, the Intra_8x8 block prediction mode has been added as part of the *Fidelity Range Extension* (Fidelity Range Extension (FRExt)) of the standard. There are nine 4x4 and 8x8 possible block prediction directions and four 16x16 block prediction directions. For the chrominance component, an MB makes use of 8x8 block prediction mode only. There are four 8x8 possible block prediction directions. The prediction directions for the 8x8 prediction mode are similar to the ones used for the 16x16 prediction mode in the luminance component.

These intra prediction modes include a directional prediction, thus greatly improving the prediction in the presence of directional structures. With the Intra frame prediction, the I pictures can be more efficiently encoded than in other standards which do not use Intra frame prediction.

For each MB, and for each color component (Y,U,V), one prediction mode and one set of prediction directions is maintained. The H.264 encoder selects the best combination mode/direction by using the Sum of Absolute Errors (SAE). This implies that for each existing direction of each mode, the predictor within the pixel domain is created from the boundary pixels of the current partition and the SAE costs are evaluated. The best combination of mode/direction is determined corresponding to the one presenting the minimum SAE cost. The residual is encoded using a 4x4 integer based transform.

Coding functions	Baseline Profile	Main Profile	Extended Profile
I slices	X	X	X
P slices	X	X	X
B slices		X	X
SP and SI slices			X
CAVLC	X	X	X
CABAC		X	
Slice Groups and ASO	X		X
Redundant Slices	X		X
Weighted Prediction		X	X
Data Partitioning			X
Interface		X	

Table 1.1: H.264 Baseline, Main and Extended Profiles

1.2.2.3 H.264 Profiles

H.264 defines a set of *Profiles*, each supporting a set of coding functions and each specifying the requirements of a decoder that satisfies the *Profile*. Table 1.1 summarizes the different options available in the three profiles defined in the H.264 standard.

In general, the *Baseline Profile* is designed for video telephony, video conferencing and wireless communications. The *Main Profile* may be useful for broadcasting media applications, such as digital television and video storage, while one potential application for the *Extended Profile* is multimedia streaming.

A video picture can be coded as such, if it has all the macroblocks of the video picture or more slices otherwise. The number of macroblocks per slice does not need to be constant within a picture. There is a minimum inter dependency between coded slices which can help to limit the propagation of errors. There are five types of coded slices shown in Table 1.2. A coded picture may be formed by different types of slices. The types of slices available depend on the profile selected.

In the following sub-sections the different profiles available in the H.264, the coding functions and the slice types are briefly described. Nevertheless, the interested reader can find more information related on this topic in [61].

Slice Type	Description	Profiles
Intra (I)(Intra)	Contains only I MBs	All
Predicted (P) (Predicted)	Contains P and/or I MBs	All
B! (**B!**) (Bi predictive)	Contains B and/or I MBs	Extended and Main
Switching P (SP) (Switching P)	Facilitates switching between coded streams: contains P and/or I MBs	Extended
Switching I (SI) (Switching I)	Facilitates switching between coded streams: contains SI, a kind of I MBs	Extended

Table 1.2: H.264 slice mode

1.2.2.3.1 The Baseline Profile

The *Baseline Profile* supports coded bit-streams containing I and P slices. P slices can contain Intra, Inter or skipped macroblocks. If one macroblock is encoded as skipped, no more data are sent to that macroblock. Inter MBs are predicted using previously coded pictures, using motion compensation with quarter sample motion vector accuracy (in the luminance component). The use of an H.264 encoder capable of inserting a picture delimiter RBSP unit at the boundary between coded pictures is recommended. This shows the start of a new coded picture indicating which slice types are allowed in the following coded picture. If this mechanism is not used, the decoder will expect to detect the occurrence of a new picture based on the header of the first slice in the new picture.

Other options available in the *Baseline Profile* are resumed in the following lines:

- *Redundant slices.* The encoder can encode redundant pictures, within the full or with part of the coded picture. These pictures will be used in case the primary coded picture is damaged during transmission or storage.

- *Arbitrary Slice Order* (ASO). The slices in a coded frame may follow any decoding order.

- *Slice groups.* A slice group is a subset of the macroblocks in a coded picture and may contain one or more slices. Within each slice in a slice group, MBs are coded in raster order. If only one slice group is used per picture, then all the MBs in the picture are coded in raster order. In this case, ASO can not be used.

1.2.2.3.2 The Main Profile

In general, the *Main Profile* is a superset of the *Baseline Profile* where B slices (bi-predicted), weighted prediction for creating a motion-compensated prediction block, interlaced video (frames or fields) and *Context-base Adaptive Binary Arithmetic Coding* (CABAC) as entropy coding method, are mechanisms enhancing the Baseline Profile. These mechanisms are optional; they can be enabled or disabled in the H.264 standard. However, in this profile the redundant slices, ASO and multiple slice groups are not supported.

A B slice may be predicted from one or two reference pictures, before or after the current picture in temporal order. It depends on the reference pictures available in the encoder and decoder. In this way, there are more options to select the prediction reference for the macroblocks in a B slice. Macroblock partitions in this kind of slice can be done in direct mode, motion-compensated or motion-compensated bi-predictive. The different algorithms proposed in this dissertation only run with I and P slices, reason for which no more details will be provided on this kind of slices.

Weighted Prediction is a method of modifying the samples of motion-compensated prediction data in a P or B slice macroblock. The prediction samples may be scaled by a weighting factor, before obtaining the motion compensated prediction. A large weighting factor is applied if the reference picture is temporally close to the current picture and a smaller factor is applied if the reference picture is temporally far away from the current picture. This tool may be useful when the sequence has *fade transitions*, where one scene fades into another.

Another functionality available in the *Main Profile* is the interlaced video. The encoder can choose to encode each MB pair as two frame MBs or two field MBs and may select the optimum coding mode for each region of the picture. Coding a slice or macroblock pair in field mode requires modifications to a number of encoding and decoding steps. All the coded fields are treated as separate reference pictures for the P or B slice prediction. The prediction of coding modes in Intra macroblocks and motion vectors in Inter macroblocks require modification, depending on whether adjacent macroblocks are coded in frame or field mode.

The CABAC [63, 64], achieves good compression performance by selecting probability models for each syntax element according to the element's context, adapting probability estimates based on local statistics and using arithmetic coding rather than variable length coding. The definition of the decoding process is designed to facilitate low complexity implementations of arithmetic encoding and decoding. Besides, CABAC provides improved

coding efficiency compared with VLC. The arithmetic operations for implementing the CABAC are described in the H.264 standard decoder [60].

1.2.2.3.3 The Extended Profile

The *Extended Profile* focuses on video streaming applications. As shown in Table 1.1, it includes all the *Baseline Profile* characteristics. The new features focus on supporting efficient streaming over packet switched networks, error resilience and noise environments.

SP and SI slices allow efficient switching between video streams and random access for the video decoders [65]. Over the Internet, where the data throughput may drop suddenly, the decoder can switch automatically between the same sequence encoded using different bit rates. This is the function of the SP slices. They are designed to support switching between similar coded sequences. For example, the same sequence at different bit rates. In this case, the motion compensated prediction may be very efficient. This solution is better than inserting I frames at switching points, improving the performance too. Besides, SP slices allow random access features. On the other hand, SI slices are used to pass between one sequence to a completely different sequence, in which case it will not be useful to use motion compensated images, because there is no relationship between them. More detailed treatment of the process can be found in [66].

The *Data Partitioned Slices* is a feature designed to improve the robustness of the transmission of an H.264 encoded sequence. The coded data of a slice are distributed into three different partitions, each of them containing a subset of the data. The first one has the header of the slice and the header of the data for each macroblock. This partition is highly sensitive to transmission errors. The second partition contains the residual data for the Intra and SI slice macroblocks and the last one contains the coded residual data for Inter coded macroblocks, forward and bi-directional. The data of each partition can be placed in a separate NAL unit, i.e. they can be stored or transmitted separately. If some data from the two last partitions are lost, the sequence may be decoded and part of the missed information can be reconstructed.

1.2.3 Video coding standard: HEVC

Recently, the ISO/IEC Moving Picture Experts Group (MPEG) and ITU-T Video Coding Experts Group (VCEG) established a Joint Collaborative Team on Video Coding (JCT-VC) to develop the High Efficiency Video Coding (HEVC) standard. The technical content of HEVC was finalized on

January, 2013 and the specification was formally ratified as a standard on April 2013.

HEVC is a video compression standard, a successor to H.264/MPEG-4 Advanced Video Coding (AVC), that was jointly developed by the ISO/IEC Moving Picture Experts Group (MPEG) and ITU-T Video Coding Experts Group (VCEG) as ISO/IEC 23008-2 MPEG-H Part 2 and ITU-T H.265.[1][2][3][4] MPEG and VCEG established a Joint Collaborative Team on Video Coding (JCT-VC) to develop the HEVC standard.[1][2] The technical content of HEVC was finalized on January 25, 2013 and the specification was formally ratified as a standard on April 13, 2013. The second version of the standard was completed in July 2014, and is expected to be published in late 2014.

HEVC is a video compression standard, a successor to H.264/MPEG-4 AVC (Advanced Video Coding) with the aim to significantly improve the compression efficiency compared with the existing H.264/AVC high profile.

In this section we will review only the most relevant differences that HEVC includes with respect to H.264/AVC, for a detailed overview pleas refer to [67, 68].

Some of the key elements of the HEVC test model are:

- A more flexible block structure with block sizes ranging from 64x64 down to 8x8 pixels using recursive quad-tree partitioning.
- Improved mechanisms to support parallell encoding and decoding.
- More intraprediction modes, up to 35, directional and supporting several block sizes.
- Support for several integer transforms, that can be applied to transform blocks ranging from 32x32 to 4x4 pixels.
- Improved motion information.

1.2.3.1 Picture partitioning

HEVC is a block-based hybrid-coding scheme. One of the major contributions to the higher performance of HEVC is the introduction of larger block structures with flexible subpartitioning mechanisms.

The basic block is known as Larger Coding Unit (LCU) and each picture is partitioned in LCU upt o 64x64 pixels each one.

LCUs can be recursively split into smaller Coding Unit (CU) which are used as the basic unit for intra- and intercoding and can have the size of the LCU or being recursivelly partittioned up to a size of 8x8 pixels.

Each CU can be in turn further split into Prediction Unit (PU), which form the basis for prediction. Each CU may contain one or more PUs in a non-recursive partitioning schema, and each PU can be as large as their root CU or as small as 4x4 pixels in luma block size. and Tansform Unit (TU). The CU partitioning in PUs can be symetric or asymmetric. Symetric PUs must be square or rectangular and are used in intra- and interprediction. Asymetric PUs are always rectangular allowing to match the boundaries of objects, and are used only in interprediction.

Each CU can also be also split in one or more transform units (TU), which is the baisc unit for transoform and quantization processes. The size and the shape of the TU depend on the size of the PU. The size of square-shape TUs can be as small as 4 x4 or as large as 32x32 and can be split in a quad-tree segmentation structure. Non square TUs can have sizes of 32x8, 8x32, 16x4, or 4x16 luma samples.

In H.264 the block picture partitioning schema is much more rigid than in in HEVC, and may not be well suited for all kinds of image content. Large blocks will generally work better for smooth regions of a picture, whereas edges and texture regions will often benefit from smaller block sizes. As the picture resolution of videos increases from standard definition to HD and beyond, the picture will contain larger smooth regions. This is the reason that HEVC supports larger encoding blocks while allowing smaller blocks to be used for more textured regions.

1.2.3.2 Slices and tiles

HEVC introduced tiles as a means to support parallel processing, with more flexibility than normal slices in H.264/AVC. Tiles are specified by vertical and horizontal boundaries with intersections that partition a picture into rectangular regions whose size could be not uniform. This offers greater flexibility and can be useful for error resilience applications. Tiles are processed in raster order, and in turn, inside a tile the LCUs are processed in a raster scan order too.

HEVC also supports slices, similar to slices in H.264/AVC, but without Flexible Macroblock Ordering (FMO). Slices and tiles may be used together within the same picture. To support parallel processing, each slice in HEVC can be subdivided into smaller slices called entropy slices. Each entropy slice can be independently entropy decoded without reference to other entropy slices.

Therefore, each core of a Central Processing Unit (CPU) can handle an entropy-decoding process in parallel.

Tiles and slices produce a performance reduction since prediction dependencies are broken across boundaries and the statistics used in entropy coding have to be initialized for every slice/tile.

1.2.3.3 Wavefront processing

To avoid the performance reduction include by the use of tiles and/or slices, Wavefront Parallel Processing (WPP) is supported in HEVC. The basic concept is to start processing (either encoding or decoding) a new row of LCUs with a new thread as soon as two LCUs have been processed in the row above. Two LCUs are required because intraprediction and motion vector prediction depend upon data from both the LCU directly above the current one and the one above the right.

1.2.3.4 Intraframe coding

HEVC follows the basic idea of H.264/AVC intraprediction but makes it far more flexible. HEVC has 35 luma intraprediction modes compared with nine in H.264/AVC. Furthermore, intraprediction can be done at different block sizes, ranging from 4x4 to 64x64 (whatever size the PU has). The number of supported prediction modes varies based on the PU size. HEVC also includes a planar intraprediction mode, which is useful for predicting smooth picture regions. In planar mode, the prediction is generated from the average of two linear interpolations.

Mode dependent intrasmoothing (MDIS) is used for some intramodes to improve the performance of intraprediction. MDIS involves applying a simple low-pass finite impulse response filter to the samples being used for prediction. This smoothing of the reference signal improves the prediction performance for large PUs. in directional modes except horizontal and vertical modes.

1.2.3.5 Interprediction, variable PU size

Each PU has a set of motion parameters, which consists of a motion vector, a reference picture index, and a reference list flag. CUs can use symmetric and Asymmetric Motion Partitions (AMP)s. AMPs allow for asymmetrical splitting of a CU into smaller PUs, which improves the coding efficiency since it allows PUs to more accurately conform to the shape of objects.

1.2.3.6 Motion parameter encoding and skip mode

Motion vectors (MV) can be predicted either spatially or temporally. Furthermore, HEVC introduces a technique called motion merge.

For each PU, the encoder can choose between using explicit encoding of motion parameters, or using motion merge mode, or using the improved skip mode.

Motion merge mode involves creating a list of previously coded neighboring PUs (called candidates) for the PU being encoded. The candidates are either spatially or temporally close to the current PU. The motion information of the selected candidate is used and so, only the index of a candidate in the motion merge list is encoded.

In the new skip mode in HEVC, the encoder also encodes the index of a motion merge candidate, and the motion parameters for the current PU are copied from the selected candidate. This allows areas of the picture that change very little between frames or have constant motion to be encoded using very few bits.

1.2.3.7 Transform and quantization

HEVC applies a DCT-like integer transform on the prediction residual. HEVC includes transforms that can be applied to blocks of sizes ranging from 4x4 to 32x32 pixels. HEVC also supports transforms on rectangular blocks. The integer transforms used in HEVC are better approximations of the DCT than the transforms used in H.264/AVC. The basis vectors of the HEVC transforms have equal energy, so there is no need to compensate for the different norms, as in H.264/AVC. HEVC also incorporates a 4 x4 discrete sine transform (DST), which is used for blocks coded with some directional intraprediction modes. When using intraprediction, the pixels close to the ones used for prediction will be predicted more accurately than the pixels further away. Therefore, the residuals will be larger for pixels away from the predicted one. DST is better at encoding these kinds of residuals.

1.2.4 Wavelet based video encoders

The first attempts to use Subband Coding, and in particular Wavelet Transform (WT), in video coding date back to late 80s [69]. It is quite easy to extend the WT to three-dimensional signals: it suffices to perform a further wavelet filtering along the time dimension. However, in this direction, the

video signal is characterized by abrupt changes in luminance, often due to objects and camera motion, which would prevent an efficient de-correlation, reducing the effectiveness of subsequent encoding. In order to avoid this problem, MC is needed. Anyway, it was soon recognized that one of the main problems of WT video coding was how to perform MC in this framework, without falling again into the problem of closed loop predictive schemes, which would prevent exploiting the inherent scalability of WT.

Actually, in such schemes as [69, 70, 37] three-dimensional WT is applied without MC: this results in unpleasant ghosting artifact when a sequence with some motion is considered. The quality objective is just as well unsatisfactory. The idea behind Motion Compensated WT is that the low frequency subband should represent a coarse version of the original video sequence; motion data should inform about object and global displacements; and higher frequency subbands should give all the details not present in the low frequency subband and not caught by the chosen motion model as, for example, luminance changes in a (moving) object.

A first solution was due to Taubman and Zakhor [44], who proposed to apply an invertible warping (or deformation) operator to each frame in order to align objects. Then, they perform a three-dimensional WT on the warped frames, achieving temporal filtering which is able to operate along the motion trajectory defined by the warping operator. Unluckily, this motion model is able to effectively catch only a very limited set of object and camera movements. It has been also proposed to violate the invertibility in order to make it possible to use a more complex motion model [71]. However, preventing invertibility makes high quality reconstruction of the original sequence impossible.

A new approach was proposed by Ohm in [72, 73], and later improved by Choi and Woods [74] and commonly used in the literature [75]. They adopt a block-based method in order to perform temporal filtering. This method can be considered as a generalization of the warping method, obtained by treating each spatial block as an independent video sequence. In the regions where motion is uniform, this approach gives the same results as the frame-warping technique, as corresponding regions are aligned and then undergo temporal filtering. On the contrary, if neighboring blocks have different motion vectors, we are no longer able to correctly align pixels belonging to different frames, since ”unconnected” and ”multiple connected” pixels will appear. These pixels need special processing, which does not correspond anymore to the subband temporal filtering along motion trajectories. Another limitation of this method is that motion model is restricted to integer-valued vectors, while it has long been recognized that sub-pixel motion vectors precision is remarkably beneficial.

A different approach was proposed by Secker and Taubman [76, 77, 78, 79] and, independently by Pesquet-Popescu and Bottreau [80]. This approach is intended to resolve the problems mentioned above, by using Motion Compensated Lifting Schemes (Motion Compensated Lifting Schemes (MC-ed LS)). As a matter of fact, this approach proved to be equivalent to applying the subband filters along motion trajectories corresponding to the considered motion model, without the limiting restrictions that characterize previous methods. The MC-ed LS approach proved to have significatively better performances than previous WT-based video compression methods, thus opening the doors to highly scalable and performance-competitive WT video coding.

Chapter 2

Objective Quality Assessment Metrics

Contents

2.1 Introduction

In past years, the development of novel image and video coding technologies has spurred interest in developing digital video communications. The definition of evaluation mechanisms to assess video quality plays a major role in the overall design of video communication systems.

As [81] explains, the image quality measurement is very important for most image processing applications. An image quality metric has mainly three kinds of applications:

1. It can be used to monitor image quality like, for example, in an image and video acquisition system that can use the quality metric to monitor and automatically adjust the system to obtain the best quality. Or, a network video server can also use it to examine the quality of the digital video transmitted and control the video streaming.

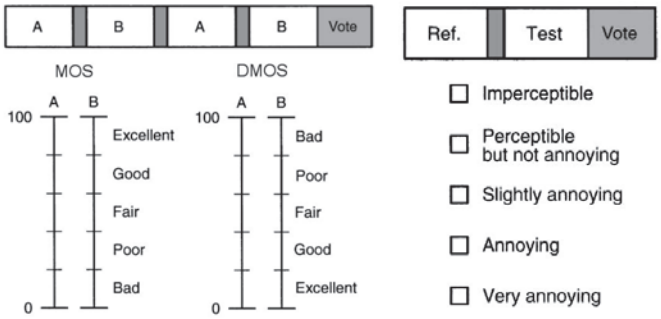
2. It can be also employed to benchmark image processing systems, algorithms, and encoder proposals.

3. And it can be embedded into an image processing system to optimize the algorithms and the parameter settings. For instance, in a visual communication system, a quality metric can help optimal design of the prefiltering and bit assignment algorithms at the encoder and the postprocessing algorithms at the decoder.

The most reliable way of assess the quality of a video or image is subjective evaluation, because human beings are the ultimate receivers in most applications. But this way of assess image quality is not appropriate for the mentioned applications.

The Mean Opinion Score (MOS), which is a subjective quality metric obtained from a number of human observers, has been regarded for many years as the most reliable form of quality measurement. However, in order to achieve statistically relevant results, the MOS method has to evaluate a huge test population, so it is too cumbersome, time consuming, not suited for real-time, and is expensive for most applications.

The MOS is generated by averaging the results of a set of subjective tests, where a number of viewers rate the image or video quality of the presented images or sequences by way of one of the standardized methodologies proposed in the following international recommendations:



(a) (b)

Figure 2.1: Presentation sequence and rating scale for (a) DSCQS (b) DSIS, methods.

- International Telecommunication Union Recommendation (ITU-R) BT.500-11 (2002) & ITU-R BT.500-12 (09/2009) [82, 83] Methodology for the subjective assessment of the quality of television pictures: This recommendation provides methodologies for the assessment of picture quality including general methods of test, the grading scales and the viewing conditions. It recommends the Double-Stimulus Impairment Scale (DSIS) method and the Double-Stimulus Continuous Quality-Scale (DSCQS) method, as well as alternative assessment methods such as Single-Stimulus (SS) methods, stimulus-comparison methods, Single Stimulus Continuous Quality Evaluation (SSCQE), and Simultaneous Double Stimulus for the Simultaneous Double Stimulus for Continuous Evaluation (SDSCE) method.

- ITU-T P.910 (04/2008) [84] Subjective video quality assessment methods for multimedia applications: These describes non-interactive subjective assessment methods for evaluating the one-way overall video quality for multimedia applications such as videoconferencing, storage and retrieval applications, tele-medical applications, etc.

The three classes of subjective assessment methodologies: single stimulus methods, comparison methods, and double stimulus methods, recommended in these standards, are briefly summarized below.

- Double Stimulus Continuous Quality Scale (DSCQS): The reference and the distorted image (or sequence) are presented twice to the viewer, alternating between reference and distorted versions, see Figure 2.1(a). The

viewers should rank the perceived quality in a continuous scale of 0-100 (being 0 bad and 100 excellent). Multiple pairs of reference and distorted images (or sequences) are shown to the viewers, but they are not told which one is the reference or the distorted one. Analysis is based on the difference in rating for each pair, which is often calculated from an equivalent numerical scale from 0 to 100. In the case of DSCQS, the Difference Mean Opinion Score (DMOS) could be used instead of MOS. It consists of the mean of differential subjective scores. For each viewer and image (or sequence), the raw scores are first converted to difference scores; that is, the difference between the given score to the reference and distorted version. These scores are further normalized as explained in [85] to obtain Zscores [86] that are finally rescaled to the 0-100 range to obtain the DMOS score for that image or sequence, where 0 represents the best quality value (no difference between reference and distorted image).

- Double Stimulus Impairment Scale (DSIS): Unlike DSCQS, the viewers know which one is the reference image (or sequence), that is presented first, followed by the distorted one. In DSIS variant II, this presentation is repeated once. The viewers rate the images/sequences in the five-level scale presented in Figure 2.1. This method is named as Degradation Category Rating (DCR) in the ITU-T P.910.

- Single Stimulus Continuous Quality Evaluation (SSCQE): Here, the viewers are only shown the distorted image/sequence, but for a longer duration than in the previous methods, typically 20-30 minutes, and rate simultaneously while watching the sequence the perceived quality using a slider on the same scale as DSCQS.

- Absolute Category Rating (ACR): Like SSCQE, this is a single stimulus with only the distorted version shown to the viewers. They provide a single quality rate for the overall sequence using the five-level scale from Figure 2.1(a).

- Pair Comparison (PC): This method pairs the references and distorted versions in any possible combination of compression degree and/or used encoder. The pair is shown twice in rapid succession and at the end the viewer should choose which version of the pair has better quality.

These methods generally have different applications. DSCQS is is the preferred method when the quality of test and reference sequence are similar, because it is quite sensitive to small differences in quality. The DSIS method is better suited for evaluating visible impairments clearly, such as artifacts caused by transmission errors, for example. As for all subjective tasks, different results can be achieved depending on how the video or image content

is presented to the viewers and which method is used. Additionally, for the same content, the way, and the order in which it is presented to the viewers can bias the results in a desired direction.

In [87, 88, 89] authors review and compare some of these standardized testing methodologies, emphasizing the benefits and problems of each method. They analyzed the results of SSCQE and DSCQS methods, concluding that high correlated results between these methods can be achieved if the SSCQE duration of the sequences is reduced to 9 to 15 seconds. Their experiments conclude that the participating viewers considered at most the last 9 to 15 seconds of video when forming their quality estimate. This is not to say that long sequences are completely without other merits. Nonetheless, when long video sequences are used in SSCQE tests, test designers should not necessarily expect a panel of viewers to rate the video inherently differently than if shorter sequences are used. The advantages of using SSCQE as a substitute of DSCQS for video comparisons, would include faster testing (or more clips rated for the same amount of viewing time spent) and less viewer fatigue.

Another comparison of the DSCQS and DSIS II scales can be found in [90, 89], where authors study the effects of context in the different methods. One type of contextual effects is created when there are fluctuations in the subjective rating of sequences based on the types and amount of impairments presented in the preceding sequence in the test. For example, a sequence with moderate impairment that follows a set of sequences with weak impairment may be judged lower in quality than if it follows sequences with strong impairment. A common method used to try and counterbalance this type of contextual effect is the randomization of the test trial presentation order. Using it, they finally conclude that the DSCQS method has reduced contextual effects, being the best method to use in order to minimize contextual effects for subjective picture quality assessment.

These aforementioned studies reveal that the selection of the proper method for presenting the references and the distorted versions of our images or sequences could result in varying results. Besides, we have to take into account the time needed to prepare the test images, the distorted versions, the ordering of the test sequences, the viewing conditions, and to be able to enroll sufficient viewers to have statistically representative results.

Traditionally, in order to avoid the need to perform such time consuming subjective tests, the scientific community has mostly used the Mean Square Error (MSE) and the Peak to Noise Ratio (PSNR) to assess quality and compare the performance of different and competing encoding proposals. This is because MSE and consequently PSNR have many attractive features [1], they are simple to calculate and parameter free, mathematically easy to deal

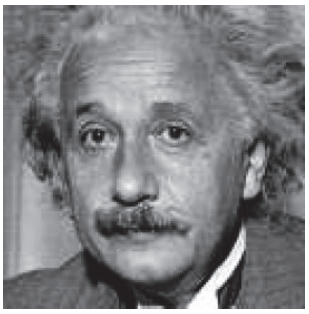
for optimization purposes, are the natural way to define the energy of the error signal, and finally is the most commonly used metric. Technically, MSE measures image difference, whereas PSNR measures image fidelity, i.e., how closely an image resembles a reference image, usually the uncorrupted original. Due to the popularity of these metrics, most of the results from previous comparison works are expressed with them, because using them saves time and effort while comparing, and as a side effect, they further propagates the use of MSE and PSNR.

In relation with human perception, MSE and PSNR are widely criticized [91, 89, 92, 93]. PSNR does not always agree with the evaluations of the Human Visual System (HVS); therefore when it is used to predict, or correlate results, with human perception of fidelity and quality, it seems not to be the best choice. The human eye, for example, does not observe small changes of intensity between individual pixels, but is sensitive to the changes in the average value and contrast in larger regions. Another deficiency of these distortion functions is that they measure only local, pixel-by-pixel differences, and do not consider global artifacts, such as blockiness, blurring, jaggedness of the edges, ringing, or any other type of structural degradation of the image.

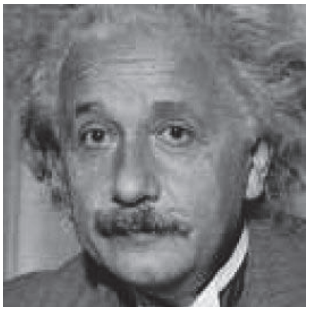
The visibility of distortions depends on the image background, a property known as masking (see section 2.3.8). Distortions are often much more disturbing in relatively smooth areas of an image than in texture regions with a lot of activity, an effect not taken into account by pixel based metrics. Therefore, the perceived quality of images with the same PSNR can actually be very different. An illustrative example is shown in Figure 2.2 where an original is altered by different types of distortions. Note that the PSNR values, relative to the original image 2.2(a) of several distorted images are nearly identical, even though the images present dramatically and obvious different visual quality. In [1], the problem with MSE is deeply studied.

But they are still the most widely used metrics in comparisons of encoder performance. This, as we will see later, can produce erroneous conclusions about the goodness of a specific encoding proposal. Nevertheless, some authors [94] argue that in scenarios with fixed content distorted by typical compression and channel artifacts, PSNR predicts the perceived subjective quality nearly as well as more complex quality models representing the state-of-the-art.

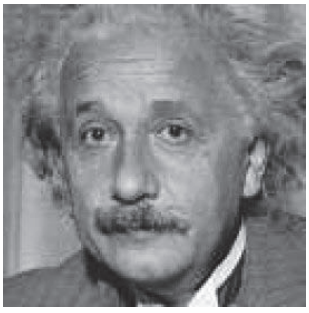
The aim of research in the field of image and video objective quality assessment is to design quality metrics that can automatically predict and rank the quality of an image or video sequence giving a quality value that is highly correlated to the subjective MOS or DMOS value given by human observers. These metrics are valuable because they provide image and video encoder designers, and standards organizations with the means for making meaningful



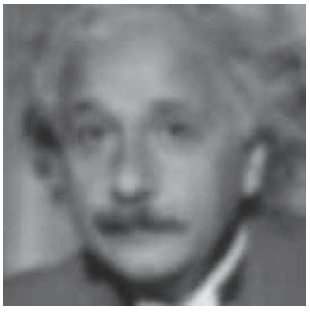
(a) Original



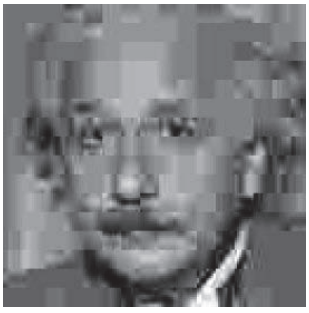
(b) PSNR=26.55



(c) PSNR=26.55



(d) PSNR=26.60



(e) PSNR=26.55

Figure 2.2: Einstein original image (a) and different distorted versions of it; The same PSNR but different perceptual quality; b) Mean Shifted Image; c) Contrast Stretched Image; d) Blurred Image; e) JPEG Compressed Image

quality evaluations without convening viewer panels, and provides big savings in time, effort, and costs.

So, one of the objective in this work is to find, among the most important image objective quality assessment metrics, one that exhibits good behavior for a large set of image (or intra-mode encoded video) distortions providing measures as close to the ones perceived by human observers and fast enough for their practical use.

In the literature, there is a consensus in a primer classification of objective quality metrics [95, 96, 89] attending to the availability of original non-distorted info (the reference) to measure the quality degradation of an available distorted version:

- Full Reference (FR) metrics perform the distortion measure with a full access to the original version, which is taken as a perfect reference.
- No Reference (NR) metrics have no access to reference. So, they have to perform the distortion estimation only from the distorted version. In general, they have lower complexity but are less accurate than FR metrics and are designed for a limited set of distortions.
- Reduced Reference (RR) metrics work with some information about the reference (similar to a perceptual hash algorithm). A RR metric defines what information has to be extracted form the reference, so it can be compared with the same information extracted from the distorted version. This reference side information is the only information available to the metric to perform the quality assessment.

The most widely used FR objective video quality metrics by the scientific community, as mentioned before, are MSE and PSNR. In recent years, new objective image and video quality metrics have been proposed, mostly for FR/RR quality assessment. They emulate human perception of image/video quality since they produce results that are very similar to those obtained from subjective methods.

Most of these proposals were tested in the different phases carried out by the Video Quality Experts Group (VQEG), which was formed to develop, validate, and standardize new objective measurement methods for video quality. The models provided by VQEG forum result in International Telecommunication Union (ITU) recommendations and standards for objective quality models for both television and multimedia applications [97].

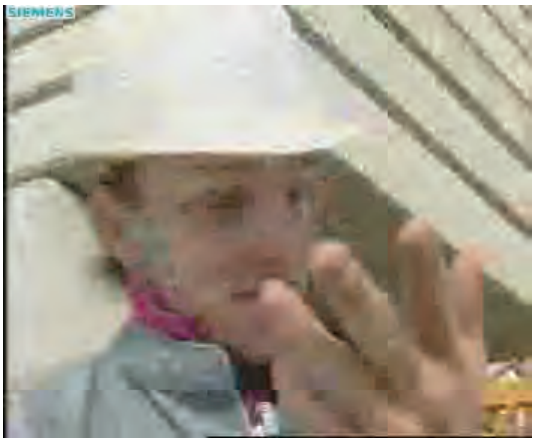


Figure 2.3: Artifacts: Blockiness

Figure 2.4: Artifacts: Blur

2.2 Principal coding artifacts and visual distortions

Most of the image or video compression algorithms used in coding standards rely on the use of the DCT or the wavelet transform. In such coding schemes, the quality of the reconstructed version of the scene is deteriorated by the loss of information and by the introduction of coding artifacts. The loss of information is produced in the quantization step of the coding chain, while other artifacts can be introduced in other steps of the chain.

Evaluation and classification of image coding artifacts [98] and video coding artifacts [99] are important in order to evaluate the performance of coding software and hardware products proliferating in the telecommunications, entertainment, multimedia, and consumer electronics markets. A comprehensive classification will also assist in the design of more

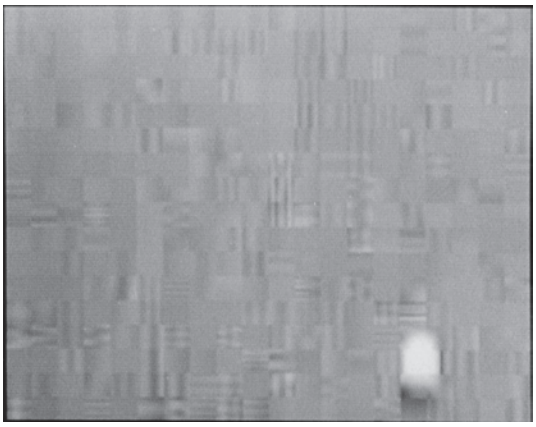


Figure 2.5: Artifacts: DCT basis image.

effective adaptive quantization algorithms and coding mechanisms in order to improve image and video codec performance. But, due to the complexity of the HVS, the perceived distortion is not directly proportional to the absolute quantization error [99].

In addition, our perceptual response to visual distortion varies depending not only where quantization errors occur, but also how they coincide with structural image elements [100]. So, it is not possible to predict the quantization level or the bit rate at which a specific artifact appears. And due to the different varieties of bit allocation techniques that have been proposed, which may, or may not, exploit the masking effects of the HVS, this prediction is even more complicated.

Nevertheless, many efforts have been made to perform adaptive quantization to reduce artifacts produced by encoders that use specific transforms, like DCT [101, 102, 103, 104] and DWT [105, 106, 107]. In addition, some specific artifacts produced by the DCT transform, like blocking, are eliminated by the use of DWT techinques [108].

The classification of coding artifacts is important too in the design of filtering and for the search of objective psychovisual-based quality metrics.

Noise and artifact are terms used to describe speckles, spikes, missing data, and other marks, impairments, defects, and abnormalities in image data created during the acquisition, transmission, and processing of image data.

The following summarizes the most common noise and artifacts produced mainly in the processing of image data, describing only how they manifest and their possible causes and relationships. Some of these effects arise only in block-based DCT schemes, others only in DWT schemes where the transform

Figure 2.6: Artifacts: Ringing on DWT.

is applied to the whole image/frame, and finally some of them arise in block-based DWT schemes like JPEG2000. For example, the transform in the LTW encoder [109] is applied to the entire image, therefore none of the block-related artifacts occur. Instead, blurring and ringing are the most prominent distortions in this type of encoders. Figures 2.3 to 2.8 show some of these artifacts.

- The blocking effect or blockiness (figure 2.3) refers to the appearance of a block pattern in the reconstructed sequence. This is due to the independent quantization of individual blocks (usually of 8x8, 16x16, etc. pixels in size) in block-based DCT coding schemes. It is more visible in low-detail regions when coarse quantization is applied to adjacent blocks, producing discontinuities at the boundaries of those blocks. The blocking effect is often the most prominent visual distortion in a compressed sequence due to the regularity and extent of the pattern. The false edges of the blocking effect are perceived as abnormal high frequency components in the spectrum of the image.

- Blurring manifests itself as a loss of spatial detail and a reduction of edge sharpness in regions with moderate and high detail (figure 2.4). Different types of blurring may occur, as motion blur due to the relative motion between elements in the scene, or out focus blur due to defocused camera or lens aberrations. Blur can be also introduced when compressing the image due to filtering and the suppression of the high-frequency coefficients by coarse quantization i.e. an image appears blurred when its high spatial frequency in the spectrum is attenuated. Blurring means that the received image is smoother than the original.

- Color bleeding is the smearing of the color between areas of strongly differing chrominance, typically near edges over flat backgrounds. It results from the suppression of high-frequency coefficients of the chroma components.

- Each of the DCT basis images has a distinctive regular horizontally or vertically oriented pattern which makes them visually conspicuous (Figure 2.5). The DCT basis image effect is prominent when a single DCT coefficient is dominant in a block. The effect is caused by coarse quantization of the AC DCT coefficients in areas of high spatial activity within a frame, resulting in the nullification of the low-magnitude DCT coefficients which are within the quantization dead-zone.

- Slanted lines often exhibit the staircase effect. This is due to the fact that DCT basis images are best suited to the representation of horizontal and vertical lines, whereas lines with other orientations require higher-frequency DCT coefficients for accurate reconstruction. The typically strong quantization of these coefficients causes slanted lines to appear jagged.

- Ringing artifacts manifest themselves in the form of ripples or oscillations around high-contrast edges in compressed images. They can range from imperceptible to very annoying, depending on the data source, target bit rate, or underlying compression scheme (Figure 2.6). Ringing is fundamentally associated with Gibbs' phenomenon and is thus most evident along high-contrast edges in otherwise smooth areas. It is a direct result of improper quantization of high-frequency, leading to irregularities in the reconstruction. Ringing occurs with both luminance and chroma components. Since the high frequency components play a significant role in the representation of an edge, coarse quantization in this frequency range (i.e., truncation of the high frequency transform coefficients) consequently results in apparent irregularities around edges in the spatial domain, which are usually referred to as ringing artifacts.

- False edges are a consequence of the transfer of block-boundary discontinuities due to the blocking effect from reference frames into the

predicted frame by motion compensation.

- Jagged motion can be due to poor performance of the motion estimation. Block-based motion estimation works best when the movement of all pixels in a macroblock is identical. When the residual error of motion prediction is large, it is coarsely quantized.
- Motion estimation is often conducted with the luminance component only, yet the same motion vector is used for the chroma components. This can result in chrominance mismatch for a macroblock.
- Mosquito noise is a temporal artifact seen mainly in smoothly textured regions as luminance/chrominance fluctuations around high-contrast edges or moving objects. It is a consequence of the varied coding of the same area of a scene in consecutive frames of a sequence.
- Flickering appears when a scene has high texture content. Texture blocks are compressed with varying quantization factors over time, which results in a visible flickering effect while watching the sequence.
- Aliasing can be noticed when the content of the scene is above the Nyquist rate, either spatially or temporally.
- Masking is the reduction in the visibility of one component (the target) due to the presence of another (the masker). There are two kinds of masking effects, luminance masking (light adaptation) and texture masking, which occur when the masker and target have similar frequencies and orientations.
- Jitter distortion occurs in video sequences due to abrupt variations resulting from asynchronous acquisition of video frames
- Jerkiness refers to the perception of still images in a video sequence resulting from frame rates that are too low.
- Frame-loss is the loss of entire frames; normally, frame-loss is produced in bursts of different duration, i.e., number of frames. When frame-loss occurs, the video codec usually repeats the last correctly received frame (frame-freeze effect) or sets a black frame. Frame-freeze is considered to be detected when its duration exceeds a certain threshold.

Another type of distortions are due to transmission errors of the bitstream over a noisy channel. When compressed video is transmitted over a packet-switched network, wired or wireless, some transport protocol like Asynchronous Transfer Mode (ATM) or the Transfer Control Protocol/Internet Protocol (TCP/IP) ensures the delivery of the bitstream. Normally, the

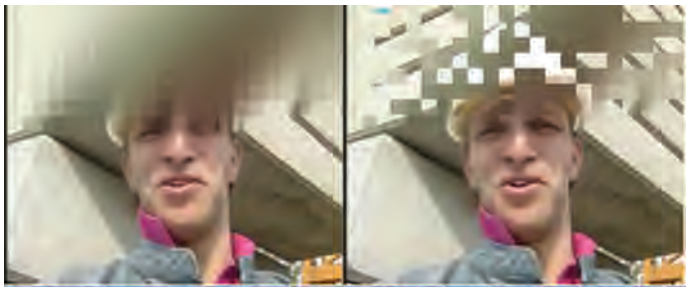


Figure 2.7: Artifacts: Two types of reconstructed frames after packet losses

bitstream is packetized, i.e., splitted in packets, whose headers contain sequencing and timing information. When the final application requires the bitstream in real-time for decoding and displaying the multimedia content, some common network conditions can produce the loss of some packets, which finally result in visual artifacts in the reconstructed sequence (figure 2.7).

In addition to the loss of packets, bit errors can occur inside packets that are not lost, producing several types of noise effects in the reconstructed image or frame (Figure 2.8) that are different depending on the codec being use and many other factors like bit allocation in the bitstream, amount of bits (burst error), importance of the bits for the coding scheme, etc.

Packets can be lost or delayed so that they are not received in time to be decoded when requested. To the decoder, both alternatives have the same effect, the packet is lost and the bitstream can not be completely decoded. If some packets need dependent information contained in the lost packets, for example, information that is differentially predicted, then the loss of a single packet corrupts the rest of the packets until the reception of the first non-dependent packet.

For example, an MPEG macroblock that is damaged through the loss of packets corrupts all following macroblocks until an end of slice is encountered, where the decoder can resynchronize. In this example, two types of errors are produced by the loss of packets: a spatial loss propagation and a temporal loss propagation. The spatial loss propagation is due to the fact that the DC coefficient of a macroblock is differentially predicted between macroblocks. The temporal loss propagation arises when the lost information is needed by motion estimation.

The visual effect of such loss depends on the ability of the decoder to deal with corrupted bitstreams. Some decoders include clever concealment techniques such as early synchronization and spatial or temporal interpolation

Figure 2.8: Artifacts: Bit Errors on DWT

in order to minimize these effects.

2.3 Brief overview of HVS

Some brief introduction to the Human Visual System (HVS) must be done in order to understand how the objective quality assessment metrics are built. Only the most important characteristics of the HVS that are implemented in these metrics are briefly reviewed here [95, 110, 111, 96, 89].

2.3.1 The visual pathway

The first contact of light with the eye is at the cornea, the main refractive surface of the eye; see Figure 2.9 from [96]. Light then enters the eye through the pupil in the center of the iris. The pupil diameter varies from 3 to 7 mm, and changes its size up to a factor of 5, based on the prevailing light level and other influences of the nervous system.

The light goes through the lens, which changes its shape with accommodation to focus the image on the back of the eye, projecting an inverted image of the visual field. After the lens, light passes through the gelatinous vitreous humor in the main body of the eye.

At the back of the eye is the retina, an extension of the central nervous

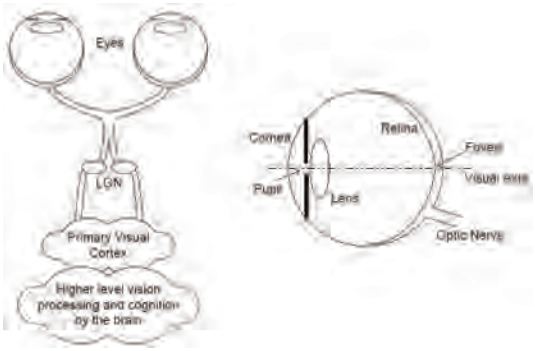


Figure 2.9: Schematic diagram of the human visual system.

system, where light sensitive photoreceptors transduce the electromagnetic energy of light into the electro-chemical signals used by the nervous system. It consists of five main neural cell types organized into cellular layers and synaptic layers.

The photoreceptors, which initiate the neural response to light, are located on the outer part of the retina. There are two classes of photoreceptors, rods and cones. The rods are responsible for vision at very low light levels (scotopic) and do not normally contribute to color vision. The cones, which operate at higher light levels (photopic), mediate color vision and the seeing of fine spatial detail, so they are responsible for vision under normal light conditions. There are three different types of cones, corresponding to three different light wavelengths. The L-cones, M-cones, and S-cones (corresponding to the long, medium and short wavelengths) split the image projected onto the retina into three visual streams. These visual streams can be thought of as the red, green and blue color components of the visual stimulus, though the approximation is crude.

The photoreceptors are non uniformly distributed over the retina. The point on the retina that lies on the visual axis is called the fovea and it has the highest density of cone cells. This density falls off rapidly with distance from the fovea. The distribution of the ganglion cells, the neurons that carry the electrical signal from the eye to the brain through the optic nerve, is also highly non-uniform, and drops off even faster than the density of the cone receptors. The net effect is that the HVS cannot perceive the entire visual stimulus at uniform resolution.

The signals from the photoreceptors are processed via retinal connections and exit the eye by way of the optic nerve. The axons of the ganglion cells, in the inner cellular layer of the retina, are gathered together and exit the eye at the optic disc, forming the optic nerve that projects to the Lateral Geniculate Nucleus (LGN), a part of the thalamus in the midbrain. These synaptic

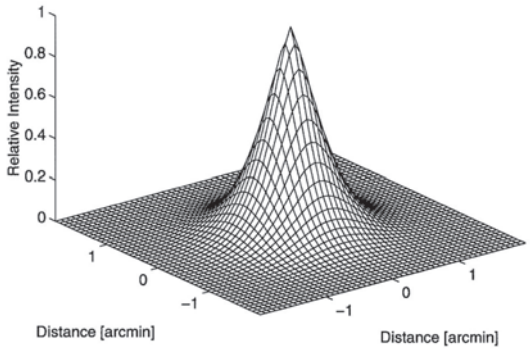


Figure 2.10: Point spread function of the human eye as function of visual angle

connections project the signal to the primary visual cortex, which contains neurons tuned to various aspects of the incoming streams, such as spatial and temporal frequencies, orientations and directions of motion. These areas in the visual cortex respond to visual stimuli and processes of various modes of vision such as form, location, motion, color, etc.

The neurons in the cortex have receptive fields that are modeled as two-dimensional Gabor functions, which are linear filters that typically are used for edge detection. The whole set of these neurons is modeled as an octave-band Gabor filter bank [112] where the spatial frequency spectrum (in polar representation) is sampled at octave intervals in the radial frequency dimension and uniform intervals in the orientation dimension. The output of these neurons saturates as the input contrast increases. The tasks of these neurons in the cortex are typically emulated in some quality assessment metrics and perceptually driven encoders, with the inclusion of models of spatial frequency and orientation selectivity.

2.3.2 Foveal and peripheral vision

As stated before, the retinal image is a distorted version of the input visual field. A natural noticeable distortion is blurring, produced by imperfections of the optics of the eye and natural variations of light produced at each step in the visual pathway.

To quantify and model the amount of blurring of a HVS, a Point Spread Function (PSF) or a Line Spread Function (LSF) is used. Its Fourier transform is the Modulation Transfer Function (MTF) of the eye for this stimulus. The amount of spreading or blurring of a stimulus is a measure of the quality of an optic system. The amount of blurring depends on the pupil size, being higher

as the pupil increases its size due to lower ambient light intensities.

This variation is modeled by a a simple formula (Equation 2.1 [95]) to approximate the foveal point spread function of the human eye with good focus and a pupil diameter of 3 mm. [113], being α minutes of arc. This PSF, presented in Figure 2.10, also changes with wavelength. By accommodation, the eye can place any wavelength into good focus, but it is impossible to focus all wavelengths simultaneously.

$$PSF(\alpha) = 0.952e^{-2.59|\alpha|^{1.36}} + 0.048e^{-2.43|\alpha|^{1.74}} \tag{2.1}$$

As commented in section 2.3.1, the densities of the cone cells and the ganglion cells in the retina are not uniform. The number of photoreceptors peak at the fovea and decreases with distance from it. Cones are concentrated in the fovea, the region of highest visual acuity, which covers approximately two degrees of visual angle on the retina. When a human observer fixates at a point of the visual scene, this point is located at the fovea being sampled with the highest spatial resolution. The surrounding points of the scene are progressively processed with lower spatial resolutions. The high-resolution vision due to fixation by the observer onto a region is called foveal vision, while the progressively lower resolution vision is called peripheral vision.

Regarding the visual spatial acuity of the fovea, the photoreceptors are packed tightly in triangular arrangement with a mean center-to-center spacing of 32 arc min. [114] This corresponds to a sampling rate of approximately 120 samples per optical degree or a Nyquist frequency of around 60 cpd (cycles per optical degree). Visual spatial acuity is therefore considered to be approximately 60 cpd, although under special conditions, for example, peripheral vision and large pupil sizes, higher spatial frequencies can also be directly resolved.

Image quality assessment models [115, 116, 117] can include foveal vision in their implementation. These models also introduce vision modeling, taking into account the non-uniform distribution off cones in the retina, modeling the image with less resolution as the distance from the region of interest (foveated part of the image) increases. Foveal vision models can resample the image with the same density of the receptors in the fovea in order to provide a better approximation of the HVS.

Most models neglect eccentricity and off-axis effects and concentrate their modeling efforts on the properties of the fovea. This is usually justified by the fact that when the eyes bring part of the image into the fovea, this part is sampled at highest resolution, being any part of the image processed in the same way. As the optical and retinal properties are relatively uniform across the

fovea, using the same properties for the whole image significantly simplifies modeling.

2.3.3 Contrast sensitivity

As commented in section 2.3.5, the HVS can perceive small differences in luminance. However, the minimal difference that still can be perceived depends on the background luminance. The dependence to the background luminance that the HVS has while detecting differences in the luminance is called contrast sensitivity. That is, sensitivity to intensity differences, is dependent on the local luminance in regions of the image [118]. A basic model for this dependence is the Weber-Fechner law. It states that, sensitivity to luminance differences in a stimulus is proportional to the mean luminance of the stimulus. Mathematically, Weber contrast can be expressed as Equation 2.2

$$C^W = \frac{\Delta L}{L} \tag{2.2}$$

The Weber-Fechner law is not fulfilled for all background luminance levels. It holds for luminance levels above approximately 10 cd/m^2 [119]; below this level the contrast threshold increases as luminance decreases, i.e., there is less sensitivity to contrast below this level. Evidently, the Weber-Fechner law is only an approximation of the actual sensory perception, but contrast measures based on this concept are widely used in vision science.

Contrast is the difference in the luminance level of adjacent parts of an image or visual field. That is, contrast is the difference in luminance or color that makes an object distinguishable. HVS is more sensitive to luminance changes (contrast) than to absolute luminance, so we can perceive objects regardless of the changes in illumination (above 10 cd/m^2 as Weber-Fechner law states) as long as the contrast is high enough.

If the contrast is too low we can not distinguish an object from the background. In this situation, some objects in the scene turn into invisible objects. These objects are said to be below the contrast threshold.

The sensitivity is the inverse of the contrast threshold, i.e., $Sensitivity = 1/threshold$. Therefore, the smaller the contrast we need to perceive an object in the scene is, the higher is our sensitivity. And the opposite, for low sensitivity we need higher contrast to perceive differences. Under optimal conditions, the contrast threshold can be less than 1%.

Suppose a scene where the contrast of an object with its background is

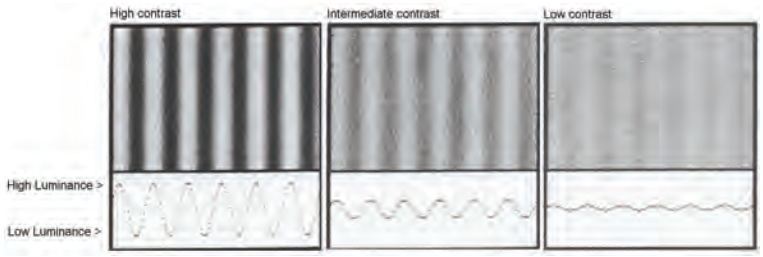


Figure 2.11: Three sine wave gratings with the same spatial frequency but with descending contrast from left to right

descending; then at just the point where the object becomes invisible we could record the value of the difference in luminance between the object and the background, this value is our contrast threshold. Its inverse is our contrast sensitivity. For example, if the contrast threshold is 0.1 then sensitivity is $1/0.1 = 10$; if the threshold is 0.01, then sensitivity is 100, and so on.

In Figure 2.11 we can see three gratings, which are called sinusoidal gratings or sine wave gratings because they change gradually in luminance over space (horizontal axis). At the bottom of each grating, a sine wave represents the luminance variability in the horizontal axis.

The contrast of periodic (often sinusoidal) stimuli with varying frequencies is defined by the Michelson contrast. The Michelson definition of contrast is in fact $(LMAX - LMIN)/(LMAX + LMIN)$, where $LMAX$ and $LMIN$ stands for Max Luminance and Min Luminance, respectively. If the sine wave of the rightmost grating in Figure 2.11 were just a horizontal line, there would be no contrast at all. Then, the so-called grating would just be a homogeneous gray, $LMAX$ would be the same as $LMIN$, and the contrast would be zero because $(LMAX - LMIN)$ would be zero. If, on the other hand, the black bars were very black and the white bars were very white, $(LMAX - LMIN)/(LMAX + LMIN)$ might be $(1000 - 1)/(1000 + 1)$, so the maximum contrast you can ever have is 1.0.

But, if in the previous scene there is more than one object and these objects are quite different in size, shape, and texture, then the point in which each object becomes invisible is different. This is due to the fact that the human perception of contrast not only depends on the difference of luminance but also on the spatial frequency. So, the contrast threshold varies with the spatial frequency.

In [120], we can find a very clear explanation of contrast sensitivity. To illustrate this, we can see figure 2.12 from [120] where three gratings are presented. Most people would rank them in the order shown, with the leftmost grating being the one with lower contrast. But this is wrong because all three

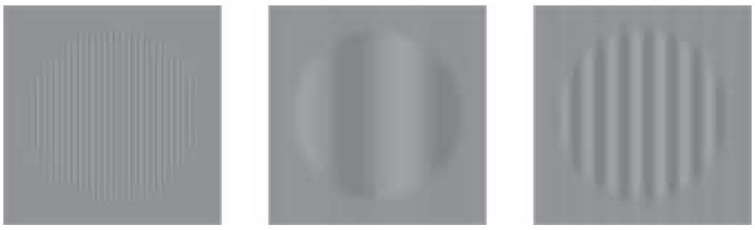


Figure 2.12: Which of these three gratings appears highest in contrast and which appears lowest in contrast?

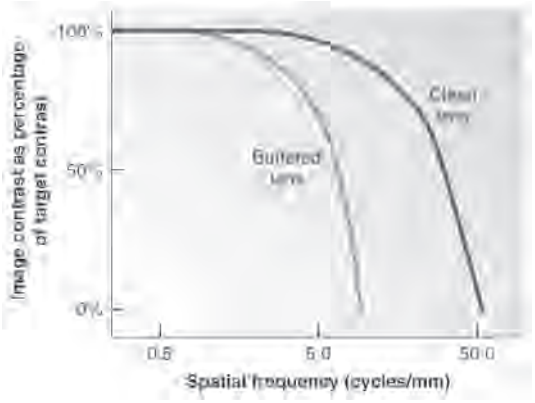


Figure 2.13: Two transfer functions for a lens. How contrast in the image formed by the lens is related to contrast in the object.

gratings have precisely the same physical contrast.

Suppose we use a lens to cast an image of a target grating on a white paper. This target grating has a specific physical contrast that we call *target contrast*. Then, using a photometer we determine the intensity of the light and dark portions in the image and, hence, the contrast of the image of the grating produced by the lens, the *image measured contrast*. We repeat these measurements for different spatial frequencies always with gratings of the same *target contrast*.

If we graph the results, whith the horizontal axis the spatial frequency of the grating and the vertical axis the *image measured contrast* as percentage of the *target contrast*, then we get the transfer function of how contrast is transferred through the lens, see Figure 2.13. In this figure two curves appear, one for a clean lens and another corresponding to a buttered lens, i.e., smeared with a buttery finger.

For the clean lens curve, the contrast in the image is identical to that of

the target up to a specific spatial frequency, but for higher frequencies the lens reproduces the target less faithfully. The frequency at which the contrast falls to zero is called the cutoff frequency; when the frequency exceeds this value, the image and the target (if a perfect lens) will no longer contain any contrast.

The curve for the buttered lens has a lower cutoff frequency, degrading the contrast of the target more rapidly than the clean lens. But at very low frequencies the smear makes little difference in the performance of the lens. This means that a high quality lens reproduces fine and coarse spatial detail better whereas a low-quality lens only reproduces low frequencies well. Think about when you are wearing smeared glasses.

Natural scenes are not as simple as gratings and that images are composed of many different spatial frequencies, sine waves in any orientation. We can treat the scene as a sum of a series of simple sinusoidal components by using Fourier analysis, we can evaluate how the lens reproduces each of those components. So we can first determine the transfer function of the lens (suppose the buttered one) and second analyze the visual scene into its spatial frequency components. Finally, with this information, we can conclude which spatial frequency components will be preserved by the lens in the image and which will not.

Suppose now that the lens is our Human Visual System: which frequencies will we perceive? The problem here is that determining the transfer function of our HVS is not as easy as with the lens.

2.3.4 The contrast sensitivity function (CSF)

With the HVS, we can not reproduce the procedure employed with the lens in order to measure the frequency components of the gratings that are preserved in the image because the image is formed inside the eye. Moreover, this image would provide information of only a part of the complete transfer function of the HVS, because other neural and cognitive components of the HVS process that image further.

As we are interested in visual perception, we must be concerned with the perceptual transfer function that depends on the optical transfer function and the neural and cognitive transfer functions. By measuring contrast thresholds for different spatial frequency gratings, we can derive a curve that describes the entire visual system's sensitivity to contrast. We call this curve the Contrast Sensitivity Function (CSF) to distinguish it from the transfer function of a lens.

Figure 2.14 shows the CSF for a human adult. The horizontal axis specifies the spatial frequency plotted as the number of cycles within a degree

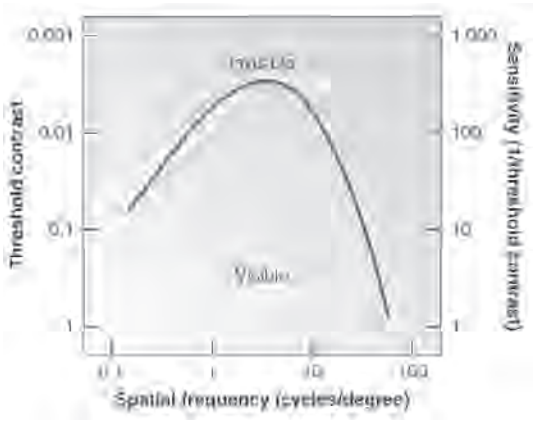


Figure 2.14: Contrast sensitivity function shape.

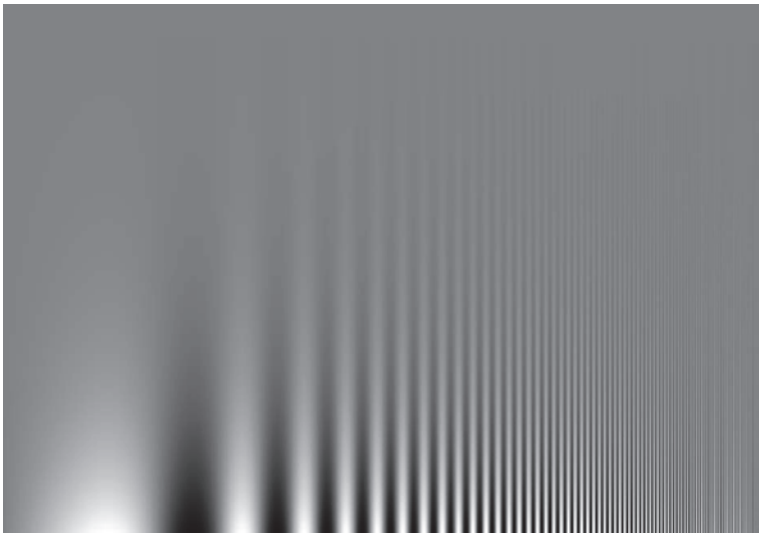


Figure 2.15: Campbell-Robson contrast sensitivity chart

of visual angle. The vertical axes plot the minimum contrast required to see the grating where the left axis show units of contrast and the right axis the inverse of this contrast value (defined as sensitivity). This curve defines the window of visibility; that is, the area underneath the curve represents combinations of contrast and spatial frequency that can be seen, while the area above represents combinations that can not be seen.

The CSF curve in Figure 2.14 differs from the lens transfer functions of Figure 2.13 at low frequencies because the HVS is less sensitive to very low spatial frequencies than it is to intermediate ones. Objects of a visual scene

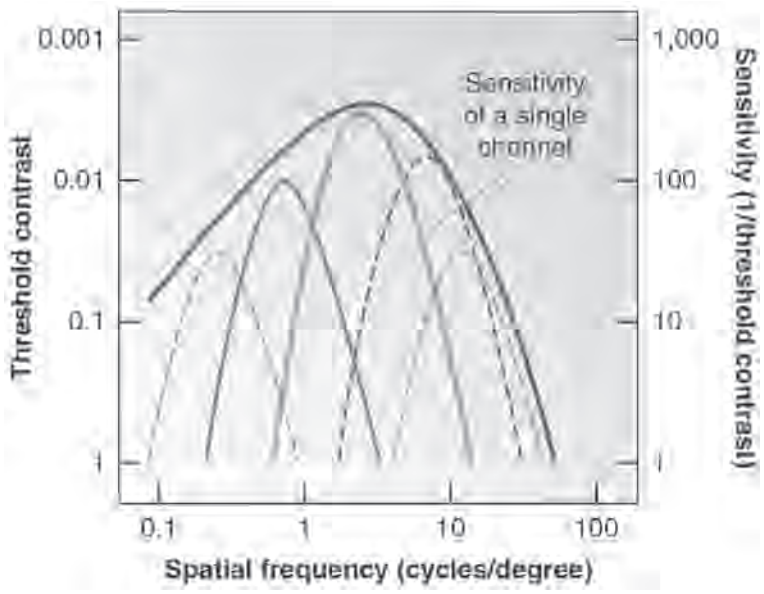


Figure 2.16: Multiple filters CSF model.

that have most of their spatial frequency information around the optimum point on the CSF will be clearly visible even when they are in low contrast. But if these objects have very low spatial frequencies (very large objects), or only very high spatial frequencies (very small objects or very fine details of them), they will be less visible and their contrast should be higher in order to be seen. This explains why the gratings in figure 2.12 appear different in contrast: their apparent contrast varies with our sensitivity to different spatial frequencies.

Figure 2.15, the so-called Campbell-Robson chart [121] demonstrates the shape of the spatial CSF for sinusoidal stimuli in a very intuitive manner. The luminance of pixels is modulated sinusoidally along the horizontal dimension. The frequency of modulation increases exponentially from left to right, while the contrast decreases exponentially from 100% to about 0.5% from the bottom to the top. The minimum and maximum luminance remain constant along a given horizontal line through the image. The location of its peak depends on the viewing distance.

Campbell [122], suggested that the CSF does not reflect the sensitivity of a single mechanism, but rather the combined activity of sets of neurons, each capable of responding to targets over only a restricted range of spatial frequencies. These independent mechanisms, called filters, detectors, or channels are responsive for detecting luminance variations that occur at a particular spatial scale (frequency). Some respond to the coarse variations and

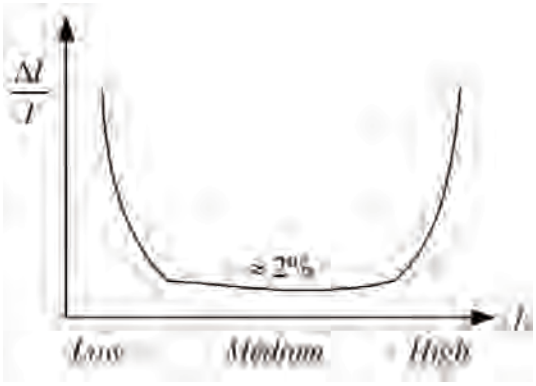


Figure 2.17: Contrast ratio: Weber fraction

others to finer details. So, the CSF reflects the envelope of sensitivities of multiple filters, see Figure 2.16. Consequently, the HVS uses the spatial frequency filters to perform a type of Fourier analysis of the retinal image.

2.3.5 CSF and light conditions

The HVS operates over a wide range of light intensity values. The scotopic and photopic vision actually cover 12 orders of magnitude, varying from the detection of a single photon to extremely bright day light conditions. To reach this dynamic range, more than a single adaptation process is involved. The first adaptation mechanism is located in the pupil, whose resizing mechanism controls the amount of light entering the eye. Then, a more powerful regulatory process of light adaptation is held in the photoreceptors and other retinal cells adjusting the gain of post-receptor neurons in the retina. The retina encodes the contrast of the visual stimulus instead of coding absolute light intensities. There are two different adaptation processes:

- Light adaptation. This adaptation happens very quickly. Sensitivity changes from dark light to bright light conditions. A decrease of the chemical concentration in the photoreceptors is the cause.
- Dark adaptation. Adaptation from bright light into darkness. In this case, the chemical concentration increases, but this process is very slow in comparison with light adaptation; it can take up to an hour until the chemical concentrations reaches its final state.

The response of the eye to changes in the intensity of illumination is nonlinear. If we consider a patch of light intensity surrounded by a

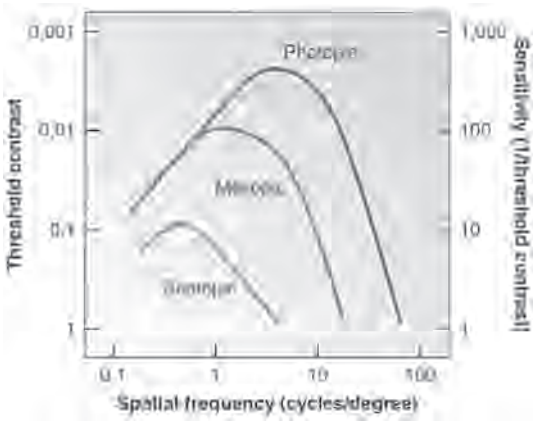


Figure 2.18: CSF under different luminance conditions

background intensity I, we can define as Just Noticeable Difference (JND) the smallest increment ΔI in luminance perceived by our HVS; [123] states that the sensitivity of human eyes to discriminate these increments depends not only on the difference itself but also on the level of intensity. Over a wide range of intensities, the Weber fraction $\frac{\Delta I}{I}$ is nearly constant at a value of about 0.02, but this result does not hold for very low or very high light intensities as shown in Figure 2.17 where $\frac{I+\Delta I}{I}$ represents the contrast ratio. So, the Contrast Sensitivity is also affected by the luminance level.

Figure 2.18 depicts how the CSF varies with light conditions, showing three CSF curves: the photopic curve (datytime), the mesopic curve (twilight), and scotopic curve (dim light). As the level of light decreases from daylight to twilight, visual sensitivity drops primarily at high spatial frequencies; this is why it is difficult to read small letters (small details) in twilight, while lower frequencies are hardly affected. When light drops further, sensitivity decreases even at low frequencies.

2.3.6 Chromatic CSF

Contrast sensitivity to chromatic spatial variations has also been studied [124] using harmonic stimuli, measuring red-green and blue-yellow gratings. Figure 2.19 from [89] shows the chromatic CSF curves in addition to the luminance CSF curve. The color CSFs are characterized as a low-pass filter with high frequencies cut-offs at much lower frequencies than the cut-off for the luminance curve. Those studies reveal that the acuity of the blue-yellow channel is limited by the distribution of the S-cones in the retina, but the red-green channel is limited by subsequent neural processing.

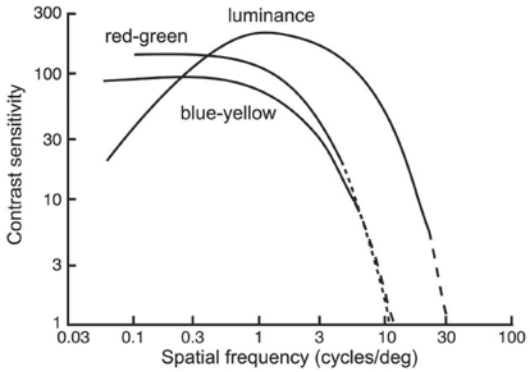


Figure 2.19: CSF for chromatic and luminance components

The sharpness of an image is judged based on the sharpness of the luminance information since the visual system is not able to solve high-frequency chromatic information. This fact has been used in the compression and transmission of color images since high frequency chromatic information can be removed without a loss in perceived image quality [124, 89]. The full range of colors is perceived only at low frequencies [125].

2.3.7 Temporal CSF

Human contrast sensitivity depends on the color, the spatial and also on the temporal frequency of the stimuli. As the spatial CSF, the temporal CSF has also a low-pass behavior. The interaction between spatial and temporal frequencies are commonly used in vision models for video [126].

Spatio-temporal CSF approximations [125] are shown in Figure 2.20. Achromatic spatio-temporal contrast sensitivity is higher than chromatic sensitivity, especially for medium-high spatio-temporal frequencies. In the achromatic chart of Figure 2.20, we can see that for low spatio-temporal frequencies, our sensitivity decreases whereas chromatic sensitivity does not. As stated before, the full range of colors is perceived at low frequencies, spatial, and temporal frequencies as shown in the chromatic chart of figure 2.20. At higher frequencies, sensitivity to blue-yellow frequencies declines first, and, at even higher frequencies, sensitivity to red-green stimuli declines too and perception becomes achromatic [125].

The space-time separability of the spatio-temporal CSF has been somewhat controversial in the literature. From a modeling and usability point of view, separability is a very interesting property in order to process video in such a

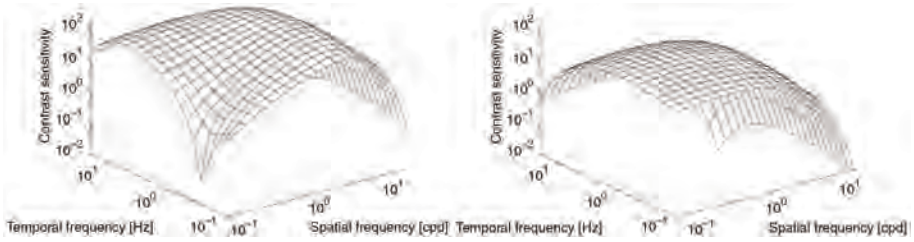


Figure 2.20: Approximations of the achromatic CSF (left) and the chromatic CSF (right)

way that takes into account the temporal dimension of the HVS sensitivity to contrast.

Early studies conclude that the spatio-temporal CSF is not space-time separable at lower frequencies [127, 128]. Further studies [129, 130] conclude that spatio-temporal CSF can be approximated by combinations of separable components in space and time. And again, later studies confirm the inseparability of space-time dimensions in the spatio-temporal CSF [131].

2.3.8 Masking

Masking is an important phenomenon in vision as it reflects the relationships and interactions between different stimuli. It occurs when a stimuli, that is visible by itself, becomes invisible in the presence of another stimuli .

There is a relationship between both stimuli, the masker, and the original stimuli. Some similar characteristics in both stimuli cause the invisibility of the original stimuli when the masker is present; normally, this interaction occurs gradually as these related properties change. These properties are the spatial frequency, the orientation, and the phase of the masker relative to the original stimuli; i.e., the masking effect is maximum when the stimulus and masker are closely coupled in terms of orientation, spatial and temporal frequency, and decreases rapidly as the distance between the signals increase in the spectral domain.

Sometimes the opposite effect, facilitation, occurs when a stimuli cause the perception of another stimuli that was not perceived before.

When talking about quality assessment, normally it is helpful to think that the distortions produced by compression, transmission, coding noise, or whatever other artifacts (original stimuli) are masked or facilitated by the image or sequence being compressed, transmitted, or coded, which acts as background.

Figure 2.21: The background image is acting as masker of a noise pattern. The original image is on the left. In the right image the noise pattern is applied to the top and bottom of the image. The texture in water and rocks makes detecting the noise pattern difficult.

Spatial masking is strongest when the interacting stimuli have similar characteristics, i.e., similar frequencies, orientations, colors, etc. But it also occurs between stimuli of different orientation and between stimuli of different spatial frequency.

For example, in some regions of the image some noise or compression artifacts are more visible than in other parts. In that cases the background image is acting as masker for the artifacts, see Figure 2.21 from [125] as an example. The noise pattern in the top part of the right image is also present in the bottom part of the same image, but the image content in this area, rocks and see, masks the noise.

So, it is important to understand which are the properties of both parts, the image in those regions, and the noise or artifact itself, because this knowledge can lead to adaptive techniques to code, compress, or transmit images in different ways in different regions.

Temporal masking accounts for the elevation of the visibility threshold due to temporal discontinuities in intensity. For example, in transitions from dark to bright the threshold elevation may last up to a few hundred milliseconds after transition.

Pattern adaptation is another type of masking that affects the contrast sensitivity due to an adjustment of the visual system sensitivity in response to a prevalent stimulation pattern [125]. Adaptation of a certain spatial frequency can lead to noticeable decrease of contrast sensitivity around that frequency.

2.3.9 Suprathreshold contrast sensitivity

Up to now, discussion was centered on at-threshold sensitivity, i.e., our sensitivity at-threshold level. Our sensitivity at-threshold is very dependent on spatial frequencies, as shown in previous sections, i.e. it depends on the spatial frequency, and thus the contrast threshold varies, having a maximum sensitivity (lower contrast threshold) in the range from 2 to 6 cpd, and as said in section 2.3.5, this varies with luminance conditions too.

When we talk about suprathreshold sensitivity, we are focusing on the visible area of the CSF (see Figure 2.14), which is the area of our regular visual conditions. There, the contrast level is above the threshold level; in other words, contrast is above the minimum level required for detecting the target over the background.

The relationship between the perception of contrast and spatial frequency at levels above threshold is slightly different than at-threshold. The effects perceived at-threshold are qualitatively different from those at suprathreshold levels, so, models of detection and discrimination levels may not be applicable because a *contrast constancy* effect (the apparent contrast matches physical contrast by an intra-channel response-gain control mechanism of the spatial frequency channels), is produced in the range from 1 to 10 cpd of spatial frequency [132, 133, 89].

The *contrast constancy* property [132] suggests that at suprathreshold levels, the contrast ratios specified by the CSF would fail to indicate veridical measures of perceived contrast; rather, the perceived contrast can be predicted based primarily on physical contrast.

The *contrast constancy* property and the effect that natural images, as masker, produce in the perception of suprathreshold targets was studied in [134] where experiments conclude that *contrast constancy* occurs only after an adaptation process and that natural images decrease the perceived contrast only of lower-frequency distortions.

In the context of lossy image compression, this *contrast constancy* property suggests that the contrasts of the distortions could be theoretically proportioned equally across the frequency spectrum (e.g., by assigning all frequency subbands equal weights) without affecting the total perceived contrast.

Because compression induced distortions are presented against natural image maskers, then, under *contrast constancy* assumption and with the support of results [134] of author's experiments, it is reasonable to assume that the post-adaptation might also affect the perceived contrast of suprathreshold distortions in a similar fashion, and as natural images decrease the perceived contrast only of lower-frequency distortions, more contrast would be allocated to these lower-frequency distortions, e.g., by assigning the corresponding subbands smaller weights (indicating less *visual importance*). Experiments in the context of lossy image compression using the wavelet transform [135] confirm too that when distortions are suprathreshold, physical contrast is a better indicator of perceived contrast than predictions based on the CSF.

The authors in [135] also detected that although *contrast constancy* is observed too for wavelet subband quantization distortions at suprathreshold levels in their unmasked experiments (without natural-images as masker), when using natural-images as masker, selective effects on the perceived contrast of low-frequency distortions are observed. The authors conclude that proportioning the contrast of the distortions according to the perceived contrast ratios, produce lower visual image quality than the one obtained by proportioning the contrast using CSF derived ratios. The authors also provide an explanation for this fact based on the global precedence mechanism, which sanctions the allocation of less contrast to lower-frequency distortion in order to preserve the visual integration of image features across scale-space.

Also in Part I of the DWT based compression standard, JPEG 2000, the *contrast constancy* property is not applied, and by the way of a visual progressive weighting factor, greater contrast allocation is given to higher-frequency distortions.

2.4 Objective quality assessment metrics

An objective quality assessment metric for images or video sequences measures the perceived distortion without human intervention in such a way that results are highly correlated to the human quality ratings for the image or sequence. It can be used as part of a quality of service monitoring application to identify changes of quality over time or as part of a rate-distortion

framework that seeks to optimize the quality of compressed images by minimizing the perceived distortion.

When comparing the performance of different coding approaches, improvements of theses approaches or completely new codec designs, the most common way of doing the comparison between proposals is in terms of the Rate/Distortion (R/D) behavior of the compared approaches. When using R/D comparisons, usually the distortion is measured in terms of Peak Signal-to-Noise Ratio (PSNR) values, while rates are often measured in bpp (bits per pixel) when comparing images or Kilobits per second (Kb/s) when comparing video sequences. However, it is well known that the PSNR metric not always captures the distortion perceived by the human being (see section 2.1).

So, a lot of efforts were performed to define objective image and video quality metrics that are able to measure quality distortion closer to the one perceived by the destination user. In this section, we perform a study of different available objective image quality metrics in order to evaluate their behavior, taking as reference the classical PSNR metric. Our purpose is to find an image quality metric that is able to substitute PSNR for image quality assessment and video quality assessment in intra mode, and substitute the PSNR as distortion metric in the R/D comparisons with that metric, thus obtaining a perceptually more accurate R/D comparison when designing and evaluating coding proposals.

The main objectives of using Quality Assessment Metrics (QAM) is to avoid the need to run a MOS test and getting the most accurate perceptual quality value of images or video sequences. An objective QAM is told to have better behavior than others if its output quality values are best correlated with the quality values given by human observers, i.e., as close as possible to the quality perceived by humans, when a MOS test is performed. Metrics for assessing how good this correlation is will be reviewed later in this section. So, QAM refers to the metrics and models for predicting this subjective visual quality scores, MOS or DMOS.

As summarized in section 2.2, many different types of distortions arise when processing, transmitting, encoding and compressing images or videos. An ideal QAM should exhibit a good behavior regardless of what kind of distortions are affecting the image. Also, it would be desirable that the time required for providing the quality measure is short enough for a practical use.

In the past years, a big effort has been made in the field of QAM. A large number or metrics can be found in the literature. Some of them have been designed for a specific kind of distortion, while others are more generalist and

try to perform regardless of the distortion type. Besides, each metric design is different. We provide a classification of image QAM. Objective evaluation of picture quality in line with human perception is still difficult [118, 91, 136, 137, 81, 138, 139] due to the complex, multidisciplinary nature of the problem, including aspects related to physiology, psychology, vision research, and computer science. Nevertheless, with proper modeling of major underlying physiological and psychological phenomena, obtaining results from psychophysical tests and experiments, it is possible to develop better visual quality metrics to replace non-perceptual criteria like PSNR or MSE.

As mentioned in section 2.1, there is a consensus in a primer classification of objective quality metrics as Full Reference, No Reference, and Reduced Reference. Most of the recently proposed image and video QAMs are Full Reference. They emulate and try to substitute the way that human perception of image and video quality is used to score the perceived quality, in the sense that they produce results that are very similar to those obtained from subjective methods. Most of the FR metrics can also provide a spatial distortion or error map for each frame or, for video sequences where they provide a time series of frame level distortion scores.

The time needed to access in FR mode both sequences is affordable for compression frameworks or applications that are not executed in real time, but not for real-time quality monitoring applications. In theses cases, NR or RR metrics are used instead. They detect classes of artifacts or error patterns in images or sequences, as blocking or blurring, but distortions for which these metrics have not been designed for remain invisible. Therefore, although most RR metrics extract features from the original image that will be compared to the same features extracted from the distorted version, there are also some RR metric that works like a FR metric but with reduced version of the original sequence. This is the case of the metric in [140, 141] that uses a low-bandwidth version of the reference for comparing with the low-bandwidth version of the distorted sequence.

The VQEG provide a forum where algorithm developers and industry users meet to plan and execute validation tests of objective perceptual quality metrics. VQEG testing includes several subjective databases whose results are to be predicted by the objective video quality models under examination. The format of the source content, the nature of the degradations, the statistical techniques and almost every aspect related to how to prepare the visual content and how to measure the results are parametrized and proposed by the VQEG. As the design of each metric provides different output quality scales, the VQEG also proposes the method to compare those heterogeneous metrics by translating the results in their own scores into a common scale to make them comparable.

Once a validation test has been completed, VQEG submits a final report to the ITU, which is ultimately responsible for preparing new standards for objective perceptual quality measurement.

VQEG has completed three validation tests. The first two tests, called Full-Reference Television (FRTV-I) [136] and Phase II (FRTV-II), covered quality measurement of standard definition television services using Full Reference models. The first test, FRTV-I , was completed in 2000. None of the models tested outperformed the PSNR. Accordingly, the initial standard, published by ITU-T Study Group 9 as Recommendation J.144 , included only informative appendices detailing objective models. The second test, FRTV-II, was completed in 2003 [137]. At the end of this validation effort, the ITU-T published an updated version of Recommendation J.144 [142] in which four objective models were included as standardized objective perceptual quality measurement methods. The third and most recent validation effort was aimed at evaluating objective perceptual quality models suitable for digital video quality measurement in multimedia applications. This project, VQEG Multimedia Phase I (MM-I), was completed in 2008 [143], and ITU-T Study Group 9 has subsequently published two new standards based on that report: ITU-T Recommendation J.247 [144] defines four new full-reference objective quality methods for multimedia, and ITU-T Recommendation J.246 [145] defines one new reduced-reference objective quality measurement method for multimedia.

2.5 QAM Frameworks

QAM can be classified by many factors, such as the metric architecture (number and type of blocks, stages or algorithms used in the metric design), the primary domain (space or frequency) where they work, the inclusion or not of HVS characteristics or HVS models in their design, and so on.

We have found different QAM reviews and different classifications [110, 125, 89, 146, 147, 93, 148, 149] in the literature , but without finding a common consensus on how to fully classify them. Some of these reviews explain with great detail most of the metrics cited here, so only the main characteristics or most relevant aspects of the metrics will be exposed here.

We grouped QAM into three different frameworks depending on the way they are designed and if their design is driven or not by any of the available HVS models.

- HVS Model Based Framework

- HVS Properties Framework
- Statistics of Natural Images Framework

If the design of one metric is not clearly based on any specific HVS model, then we move this metric out of the group of HVS modeled metrics. However, that metric can still use, somehow, one or more of the previously described HVS characteristics. The third framework is related to the statistic analysis and properties of the natural scenes.

So, in this section we will briefly describe the main ideas behind the different frameworks and the most relevant and cited QAM of each one. Normally, those main ideas are translated to functional steps or computational phases that conform the metric architecture. For each of the frameworks, we will explain briefly these phases or steps.

2.5.1 HVS model based framework

A basic idea of any metric based on an HVS model is that subjective differences between two images can not be extracted from the given images (original and distorted one), but from their perceived version. As it is known, the HVS produces several visual scene information reductions, carried out in different steps. The way in which this information reduction process of our HVS is modeled is the key to obtain a good subjective fidelity metric.

This framework includes the metrics that are clearly based on an HVS model, i.e., their design follows the stages of any of the available HVS models. We include here the Error Sensitivity Framework (ESF) [81], and also some other RR and NR metrics that are based on HVS models.

The Error Sensitivity Framework includes mainly FR metrics based on HVS models, being a common stage in all of them the quantification of the strength of the errors between the reference and the distorted signals in a perceptually meaningful way, i.e., using the HVS model. Therefore, practically all the metrics in this framework (Error Sensitivity) are FR.

Generally, the emulation of HVS is a bottom-up approach that follows the first retina processing stages to continue with different models about the visual cortex behavior, modeled as consecutive processing stages. Also, some metrics deal with cognitive issues about the human visual processing modeling that issues as additional stages.

The main difference between the FR metrics of this framework is related with the way they perform the subband decomposition inspired in the complex

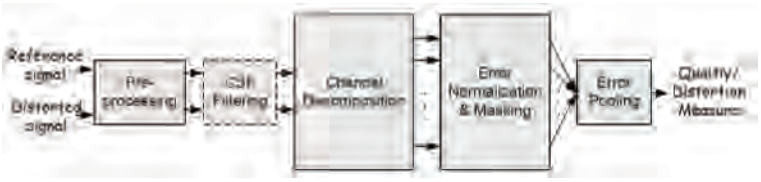


Figure 2.22: Common block diagram of the error sensitivity framework

HVS models [150, 151, 152], low cost decompositions in DCT [153, 154] or Wavelet [141] domains, and with other HVS related issues like in [155] where foveal vision is also taken into account and in [156] where focus of attention is considered. It is worth noting that a big percentage of proposed FR quality assessment models share the common error sensitivity based philosophy, see figure 2.22, which is motivated from psychophysical vision science research [96].

After some pre-processing in the space domain, the HVS usually models first decompose the input signal into spatio-temporal subbands in both the reference and distorted signal. As mentioned, this frequency decomposition is one of the biggest differences between models, and hence between metrics. Then, an error normalization, weighting process, and masking process is carried out in order to give the estimated degradation measure.

2.5.1.0.1 Pre-Processing

In this stage, some pre-processing operations are done in order to adequate some characteristic of the reference and the distorted input versions. These operations commonly include pixel alignment, image cropping, color space transformations, device calibrations, PSF filtering, light adaptation, and other operations. Not all the metrics perform all these operations; each metric adjust the inputs in a different way.

A point to point misalignment can occur due to different reasons in the compression, processing, and/or transmission of the reference image, so some metrics perform a point to point correspondence first that helps in upcoming stages to minimize assessment errors due to this fact.

Image crop is used by some authors [154, 152, 4, 157] in order to center processing in a region of interest or to avoid problems that arise in filtering stages with image boundaries. Some authors also perform some segmentation process, in order to narrow the application scope of the metric to focus in these areas. In [4], a segmentation process is done in order to determine which the *dominant blocking areas* are based on the evidence that blocking artifacts are

not noticeable likewise in all regions of the image.

Some metrics decide to convert the color signal to a space color that is better correlated to HVS. The author in [157] present a FR metric for color video sequences, based on a contrast gain control model of the HVS. He performs a conversion from the Y'Cb'Cr' space color defined in the ITU-R Recommendation 601 to an opponent color space (B-W: Black-White, R-G: Red-Green, and B-Y: Blue-Yellow) based on HVS cone sensibility to each color component. They take the behavior of conventional CRT (Cathode Ray Tube) displays into account in their color space transformations.

In [154, 152], the authors convert the reference and the distorted image into the YOZ color space, where Y is the luminance expressed in $candela/m^2$, O is an opponent color channel calculated with a specific conversion matrix, and the Z channel is the blue channel given by the International Commission on Illumination (CIE) Z coordinate. This transform also includes gamma transformation and a linear color transform.

Nevertheless, some other authors do not perform any color conversions or transformation, they in fact retain only the luminance information in order to reduce the computational cost of their proposed metrics. The authors in [4] introduce a Perceptual Blocking Distortion Metric based on the model proposed in [150]. They also perform the most important steps from the ESF, as frequency decomposition, contrast sensitivity filtering, contrast gain control, error detection, and pooling. Regarding the color conversions authors argue that only if the metric precision is a critical issue then a color conversion as in [157] is worthwhile, as it has been shown [158] that it is possible for the vision model to work on the luminance (Y) component only, without a dramatic degradation in prediction accuracy. They also propose that the contrast sensitivity band-pass filtering can be applied only to the luminance channel based on the fact that color contrast sensitivity is rather low for higher frequencies, reducing therefore computational costs.

Another type of pre-processing step is the need to convert the digital pixels (stored in the computer memory) into luminance values of pixels on the display device, through point-wise non linear transforms. Different gray-level transformations or corrections are applied as a pre-processing step in order to account for contrast adaptation to luminance conditions.

Finally, the reference and the distorted images or videos need to be converted into corresponding contrast stimuli to simulate light adaptation. There is no universally accepted definition of contrast for natural scenes. Many models work with band-limited contrast for complex natural scenes [159], which is tied with the channel decomposition. In this case, the contrast

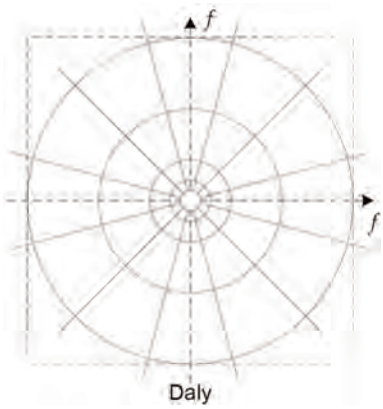


Figure 2.23: Daly frequency decomposition model.

calculation is implemented later, during, or after the channel decomposition process.

2.5.1.0.2 CSF

The CSF can be implemented in the channel decomposition step by the use of linear filters that approximate the frequency responses to the CSF like in [160] that is based on a local contrast definition and where a spatio-temporal three dimensional filter bank is applied to the image, decomposing it in different frequency perceptually channels. The filter bank design takes into account subjective psycophysical experiments in order to fix the contrast sensitivity for each frequency range and orientation, and so the frequency channel decomposition includes the contrast sensitivity function.

But most of the metrics choose to implement the CSF as weighting factors that are applied to the channels after the channel decomposition, providing a different perceptual sensitivity for each channel. In chapter 3 we will discuss how to introduce the CSF after the decomposition step but in the image encoding scope.

2.5.1.0.3 Decomposition

Transformations from the image spatial domain into the frequency domain has been extensively used in the literature in image and video coding algorithms. The most widely used frequency transforms are the Discrete Cosine Transform (DCT) and the Wavelet transform. These simple transforms have been reported due to their suitability for the codification process and certain

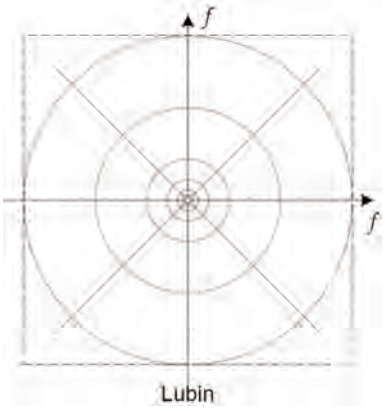


Figure 2.24: Lubin frequency decomposition model

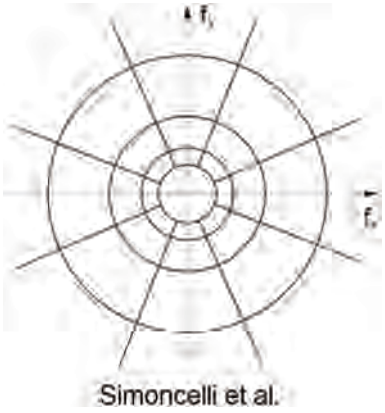


Figure 2.25: Simoncelli et al. frequency decomposition model, Steerable Pyramid.

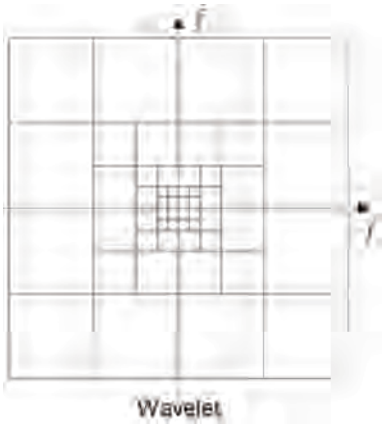


Figure 2.26: Wavelet frequency decomposition model.

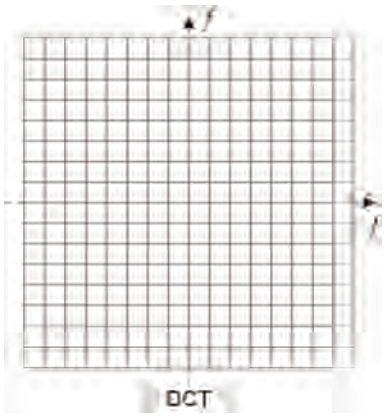


Figure 2.27: DCT frequency decomposition model.

applications, rather than their accuracy in modeling the cortical neurons; their models are not close enough to the channel decomposition as our HVS does while process the incoming signal from our eyes. Nevertheless, some metrics use a DCT [153] or wavelt [141] frequency decomposition with good correlation with MOS values.

Quality metrics that try to emulate, as accurately as possible, the way that our HVS assesses the quality of the viewed scene use more complex models of this HVS frequency channel decomposition, but taking into account the constraints of application and computation. Depending also on the metric type and the type of distortions it handles, metrics use different different channel decomposition models.

Cortical receptive fields are normally represented by 2D Gabor functions, but the Gabor decomposition is difficult to compute and is not suitable for good computational light implementation and for some operations such as invertibility, reconstruction by addition, etc.

Normally, frequency decomposition is produced by a filter bank whose design must incorporated spatial location, spatial frequency and orientation in order to resemble the HVS frequency and orientation channels. This filter bank design differs between authors. From a practical and implementation point of view, several authors have implemented pyramidal filter structures. In [161], Watson modeled frequency and orientation decomposition with similar profiles as the 2D Gabor functions but computationally more efficient. Other authors, like Lubin [115], Daly [162], Teo and Heeger [150], and Simoncelli et al. [163], provided different models trying to approximate as close as possible to the HVS channel decomposition avoiding prohibitive implementation issues. In [163], Simoncelly proposed the steerable pyramid,

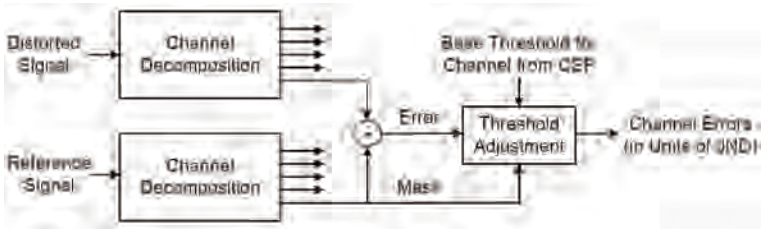


Figure 2.28: Typical implementation of masking in quality metrics.

which is a frequency multi-scale and multi-orientation image decomposition that is invariant to translations and rotations of the stimuli, without aliasing effect and invertible. In figures 2.23 to 2.27, some of these channel decomposition models are shown.

There are also some models that cover temporal frequency decompositions in order to account for the characteristics of the temporal mechanisms in the HVS [157, 160]. The design of temporal filter banks is normally implemented using Infinite Impulse Response filters (IIR) that give a delay only of a few frames; other authors use Finite Response Filters that, although having a bigger delay, are simpler to implement.

Although the use of sophisticated channel decomposition models is commonly used in QAMs, normally simpler transforms like DCT or Wavelet are still employed in the design of image or video codecs due mainly to its reduced computational cost.

2.5.1.0.4 Error Normalization and Masking

As explained in 2.3.8, masking occurs when a stimulus that is visible by itself cannot be detected due to the presence of another stimulus. Some times facilitation occurs, that is when a non visible stimulus becomes visible due to the presence of another.

Most of the HVS models in this framework implement error normalization and masking in the form of a gain-control mechanism using contrast visibility thresholds in order to weight the error signal for each channel, see figure 2.28. Some metrics [151], normally due to complexity and performance reasons, use only intra-channel masking, i.e., masking occurs only in each region of the decomposed (frequency and orientation) spectral domain, while other models [150] include inter-channel masking as there is evidence that channels are not totally independent in the HVS.

The visibility threshold adjustment at a point is calculated based on the

energy of the reference signal (or both the reference and the distorted signals) in the neighborhood of that point, as well as HVS sensitivity for that channel in the absence of masking effects (also known as the base-sensitivity). For every channel, the base error threshold (the minimum visible contrast of the error) is elevated to account for the presence of the masking signal, and for this masking elevation several masking models are typically used. The elevated visibility threshold is then used to normalize the error signal. This normalization typically converts the error into units of Just Noticeable Difference (JND), where a JND of 1.0 denotes that the distortion at that point in that channel is just at the threshold of visibility.

Some authors [164] also include in this stage the luminance masking, also called light adaptation. Detection threshold for a luminance pattern depends upon the mean luminance of the local image region. So, the brighter the background, the higher the luminance threshold. Up to a variation of 0.5 log units in the luminance threshold might be expected to occur within an image due to the mean luminance of the block for which it is calculated (assuming a block basis image encoder). Watson proposes a power function to approximate the luminance threshold for a DCT block. In [154], a local contrast calculation is included for every DCT block, converting each DCT coefficient into a value in the range from -1 to 1 that expresses the amplitude of the corresponding basis function to the average luminance in that block.

In [111, 165], we can find comparisons of different masking models and some considerations about how to include them into an image encoder. In [166], the authors propose a contrast gain-control model of the HVS that also incorporates a contrast sensitivity function for multiple oriented bandpass channels.

2.5.1.0.5 Error Pooling

Error pooling is the last step in the process, which is the process of combining the error signals in different channels into a single distortion/quality interpretation giving different importance to errors depending on the channels. For most quality assessment methods, a Lp norm or Minkowski norm is used for error pooling expressed as in Equation 2.3, where $e_{l,k}$ is the normalized error of coefficient k at frequency level l and β is a constant value lying between 1 and 4. Importance weights can also be given based on the visual importance of different regions in the image.

$$E\left(\{e_{l,k}\}\right) = \left(\sum_{l}\sum_{k}\left|e_{l,k}\right|^{\beta}\right)^{\beta} \tag{2.3}$$

Most of the previously cited metrics are FR metrics and follow the functional stages of the Error Sensitivity Framework although with variations. This schema, specifically the summation or pooling stage, allows the metrics to produce spatial error maps, frame-level distortion scores, and sequence-level distortion scores. In this sense, an image quality assessment metric can be used directly to rank video sequences. For the time domain some metrics use temporal HVS models or information to accurately reproduce human scores while others simply provide their sequence quality value as a frame-quality average.

2.5.1.1 Metrics

Now we will summarize the most relevant and cited metrics of this framework.

- In the [150] model, Teo and Heeger include basically all steps from EFS and it is one of the first reference metrics of this framework. Its model is based on the analysis of the responses of single neurons in the visual cortex of the cat, where a contrast gain control mechanism keeps neural responses within the permissible dynamic range while at the same time retains global pattern information. They perform a Quadrature Mirror Filter (QMF) frequency decomposition. The gain control mechanism is realized by an excitatory nonlinearity that is divided by a pool of responses from other neurons. The distortion measure is then computed from the resulting normalized responses by a simple squared-error norm as explained before.

- The Moving Picture Quality Metric (MPQM) [160, 151] is a FR metric that pre-processes the sequences in blocks, making a coarse segmentation of regions, uniform, pattern, and borders, in order to fix the base masking threshold for each image block. Frequency decomposition is based on a local contrast definition and Gabor-related filters for the spatial decomposition, it uses an isotropic filter for low frequencies regardless the orientation and for the frequency bands of 2,4,8 and 16 cpd and another filter for each orientation (0, $\pi/4$, $\pi/2$, and $3\pi/4$). The 17 filtered spatial decomposition are followed also by two temporal mechanisms, as well as a spatio-temporal CSF and a simple intra-channel model of contrast masking. The masking mechanisms consist of dividing the filtered error signal (original filtered minus distorted filtered) by the detection threshold obtaining data this way in *units above threshold*. Data from each channel is gathered together in a pooling step. The data provide results for a global metric and for more detailed metrics for each of the basic image components: uniform areas, contours, and textures. The global metric also

takes into account the focus of attention, computing the sequence in three-dimensional blocks, accounting for persistence of the images on the retina. Pooling this three-dimensional blocks the global distortion measure is given. The final distortion measures (global and components) can be obtained in *Visual Decibels*, expressed in the commonly used decibels decibel (dB)s or in a quality rating on a 1 to 5 scale resembling the MOS scale.

- Base on self developed non-linear and supra-threshold contrast perception model, the authors in [153] propose the use of a FR metric, working in the DCT domain that deals with a wider range of distortions than other model-based metrics. Their model is based on experimental perception results, so it models as a whole the HVS, including the effects from photoreceptors to the post-transform suprathreshold non-linearities. They argue that such a model works better than models that are based on a stage-after-stage sequential model based on disconnected characteristics of the HVS. Based on the fact that the HVS maps continuous contrast range into a finite set of discrete perceptions, they model the bit allocation properties of the HVS as a redundancy removal process analogous to vector quantization. Their experimentally parametrized Information Allocation Function (IAF) model is based on the idea that if the HVS allocated more information in one area (frequency and orientation), more visual importance is then given to that area. Their IAF value that includes not only sub-threshold or at-threshold behavior of HVS, but also the reactions to supra-threshold impairments, is used to weigh the DCT coefficients, and by measuring the differences between the perceived images (original and distorted are processed with the IAF) a subjective difference between both images is given.

- Following the ESF framework stages, in [164] Watson introduced the DCTune metric, a FR metric for monochrome images tested with the JPEG image compression standard, which was extended in [167] for color images and in [154] for color video sequences with the name of Digital Video Qualtiy (DVQ). The method treats each DCT coefficient as an approximation to the local response of a visual channel. For a given DCT quantization matrix, the DCT quantization errors are adjusted through each one of the ESF stages (contrast sensitivity, light adaptation, and contrast masking) and pooled non-linearly over the blocks of the image. This process results in a 8x8 *perceptual error matrix*, which is further pooled again for each block to give the final total perceptual error. In [164], the author argue in favor of an image dependent quantization matrix giving arguments against an image independent quantization matrix. He propose a method that, following each of the ESF stages, obtains a visually optimum

(at-threshold) quantization matrix for a specific image and bit rate. In [154], the author include the results of measurements of visual thresholds for temporally varying samples of DCT quantization noise in order to extend his previous metric to the time domain. In [152], the authors extended their previous work, providing also results from subjective tests.

- Although not following all the stages of the ESF, the authors in [87] propose a FR measuring tool for MPEG-2 video sequences. Their proposal is different as they include a *Cognitive Emulator* stage after the *Distortion Weighting* stage. This cognitive modeling of quality assessment is seldom included in quality assessment metrics, and therefore this proposal is interesting because it not only includes a low-level model of HVS, but also tries to model high-level cognitive decision stages.

  In the Distortion Weighting stage, the authors apply a low-pass filter to the original and distorted sequence with similar response to the CSF. Then, with the aid of an edge detection step that runs on the original image, a simplified masking model is applied. The masking is applied in the space domain by modifying the luminance values of the neighborhood (± 5 pixels) of the edges, with a maximum at the sharp luminance transition. The masking function is applied for vertical and horizontal edges and is composed as a combination of local masking functions for the pixels in the aforementioned neighborhood. Prior to the Cognitive Emulator, the authors obtain what they call the Instrumental Picture Quality (IPQ) . IPQ is a normalization and mapping of the PSNR to the visual rating. As subjective rating of quality saturates above and below certain quality values, they simply apply this saturation effect to the calculated PSNR of the distorted image, obtaining their IPQ this way. Their saturation limits were fixed at 20 and 50 dB. The Cognitive stage is a predictor of the subjective results from SSCQE subjective evaluation tests on video sequences. The authors propose a model to reproduce the decision making tasks involved in a SSCQE test. Their Cognitive model tries to mathematically include the biased judgment that could be expected as a result of the rapid picture quality variation in the video sequence and the need to rapidly decide the perceived quality. Based on short-term human memory behavior, the influence of strong stimuli that appears in a frame, persists during several frames. When another strong stimulus occur within an interval shorter than the memory interval this two stimuli may merge and normally mask the quality of frames inside the two distorted frames. Due to the presence of the distorted frames, the quality of the frames inside is judged to be worse than it would be in the absence of the distorted frames. This fact is modeled by the authors as a smoothing stage that modifies the IPQ value of frames between frames with a lower IPQ value. The perceptual saturation is also included in their model by

normalizing the IPQ values within the range of 0.0 to 1.0. After the Smoothing and the Perceptual Saturation stages, an asymmetric tracking stage is performed. This stage takes into account the fact that observers respond decisively and quickly to degradation in picture quality, but hesitate and slow in the case of picture improvement. They model the subjective gain and losses response by asymmetrically modifying the value of the IPQ values to account for this fact. The final stage is to delay in time the point where the modified quality value is applied in the sequence due to the human response time that was previously estimated (averaged) as 1 second. All these cognitive stages try to synchronize the video distortion with the SSCQE data.

- The author in [157] propose the Perceptual Distortion Metric (PDM), a FR metric for color video sequences, based on a contrast gain control model of the HVS. He perform a conversion from the Y'Cb'Cr' space color to an opponent color space as pre-processing stage. This metric proposes a separated temporal and spatial frequency decomposition. In the research of the temporal mechanisms in HVS, there is a consensus of the existence of at least two filtering stages, a low-pass and a band-pass referred as sustained and transient channels. Winkler uses two Infinite Impulse Response (IIR) filters to model these stages, applying the low-pass filter to all three color channels while the band-pass filter is applied only to the luminance channel to reduce complexity. The spatial decomposition is implemented with the steerable pyramid transform proposed by [163], which has the advantage of being rotation-invariant, self-invertible, and because it minimizes the amount of aliasing in subbands, but requires higher computational load. CSF is implemented as a weighting process after subband decomposition. Masking is implemented as an extension of the Watson [164] masking model to color images and to video sequences. In [158], the author tested the PDM metric with different color models. Using the CIE L*a*b* and CIE L*u*v* models with the metric has better correlation with human scores. He also concluded that using a luminance only model produced slightly lower correlations but the slight increases in accuracy of the color versions may not justify the double computational load imposed by the full-color PDM.

- Encoding images giving more bits (information) to the correct Regions Of Interest (ROI) and discarding less important information from peripheral regions can be perceptually improved by maximizing the quality value given by a foveated quality metric. Therefore, some metrics use models of the HVS that include foveation (see 2.3.2) in their design. In [155, 117], the Foveated Foveated Wavelet Image Quality Index (FWQI) is presented. FWQI is a FR metric working in the wavelet domain and based on the fact

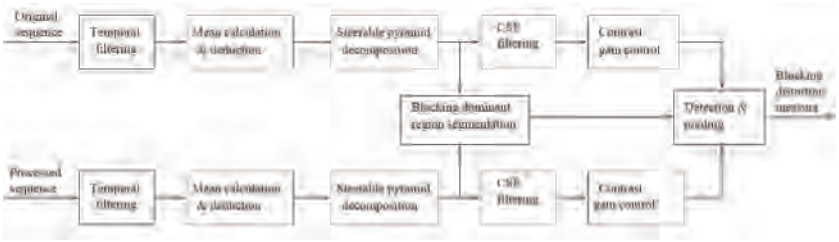


Figure 2.29: Block diagram of the PBDM [4]

that the HVS is highly non-uniform in sampling, coding, and processing. The HVS spatial resolution is higher around the fovea and decreases rapidly with increasing eccentricity. The reason of using a wavelet decomposition for this metric is because wavelet analysis delivers a convenient way to simultaneously examine frequency and spatial information. The design of this metric includes information about the space variance of the CSF, spatial variance of the cutoff frequency, and information about the variation of the human visual sensitivity in different wavelet subbands. The distance to the image and the display resolution also plays an important role. The perceptual importance of each wavelet subband is taken from the model in [106], which fixed the error sensitivity for each subband based on experimental results. The authors combine this model with a model of the distance of each wavelet coefficient to the foveation point in spatial domain, obtaining their FWQI after pooling.

- In [4], the authors propose a blocking impairment metric, the Objective Blocking Rating (OBR) and the Perceptual Blocking Distortion Metric (PBDM) based on the OBR. PDMB is a FR metric based on the [150] HVS model with the modifications made in [160] that include temporal filters and CSF, and also with the color extensions made by [157]. This extended model was finally modified to change the gain control stage to the one proposed by [166]. All the stages in the model clearly explained and slightly simplified to reduce computational effort. After some parametrization, the authors get the same correlation with MOS values as the PDM metric, but with lower computational cost.

  The main steps of [4] can be shown in 2.29. The Steerable Pyramid is used to perform the frequency decomposition, but only to a central region of the image in order to avoid boundary effects. The CSF is then implemented as a weight factor that multiplies each subband in the wavelet domain. The CSF weighted coefficients are then passed to the gain control mechanisms that square and normalize the coefficients. As is known, the *LL* subband holds the low-pass band. It is important to notice that the authors pre-process the

frames in order to be able to pass the gain-control stage to this subband by substracting the mean value to each pixel in the frame (in the spatial domain) before the frequency decomposition. This pre-processing step is needed therefore in order to prevent the accumulation of energy into the low-pass band, which could produce that the magnitude of those coefficients fall out of the the dynamic range of the gain-control stage. A final pooling stage simulates the integration process of the HVS finally obtaining the perceptual distortion map with the same size as the original frame, assigning the perceptual distortion at that spatial location to each pixel.

As shown in Figure 2.29, the authors propose and introduce an additional blocking stage so that their algorithm produces a blocking region map. They also provide a method to calculate the ringing artifacts produced after the frequency decomposition, but as ringing is produced due to edge reconstruction errors, they should not be considered as blocking artifacts so that ringing areas are excluded from the blocking region map. Both algorithms rely on experimentally adjusted thresholds. The authors averaged the summed blocking distortion by the number of frames and experimentally adjusted the dynamic range of the metric in a scale of 1 to 5. Blocking distortion is calculated in the previously segmented *blocking dominant region*.

As the viewers attention is located mainly on faces and moving objects, the authors in [156], although not proposing a novel metric, combine the use of two quality assessment metrics in order to achieve the global quality rating of a video sequences. When the focus of attention is located on a particular area of the scene, the background or the rest of objects in the scene are coarsely processed. They combine the previously commented FR PDM metric, which is based on a HVS model and NR [168] metric, to measure the influence of blockiness, blur, and jerkiness artifacts. The combined metric is guided from a semantic segmentation of images. The semantic segmentation is produced mainly for people's faces. When the focus of attention is placed on moving objects, then background objects or those with different velocities are also processed less accurately. In [126], a spatio-temporal CSF model that accounts for this is presented.

- The authors in [141], made an interesting proposal of two metrics, a FR, and a RR one for video sequences, based on the same HVS model. Their model follow all the aforementioned stages such as, color space conversion, temporal filtering, spatial filtering, contrast computation masking, and summation. As they point out, the use of a RR or a NR metric that is specifically designed for catching some impairments, like blocking or blurring, has the disadvantage of not being able to determine if one potential artifact is part of the sequence or the result of the compression process of a

new generation of codecs or algorithms. Therefore, they based their RR on the same HVS model than the FR one, but working with a reduced bandwidth version of the reference sequence. This reduction can be scaled up to FR, adapting to the available bandwidth. Although their model is based on previous HVS models, the parametrization that authors perform on the model is guided by the responses to natural video frames rather than by the responses to simple visual stimuli such as sinusoidal gratings. In addition, they propose a method to perform a perceptually driven rate control based on a previous work [169] and using the new RR metric as a distortion measure in the rate control algorithm.

2.5.2 HVS properties framework

In this framework, we include other types of metrics that although not based on a specific HVS model, are still inspired on the HVS in the sense that their design takes some of the aforementioned HVS properties into account. We also include here, those metrics that are built to detect specific impairments produced by any of the processing stages of images and videos, like quantization, encoding, transmission etc., by analyzing different image properties.

2.5.2.1 Metrics

- The Institute for Telecommunication Sciences (ITS), proposed an objective video quality assessment system that was based on human perception in [170]. Instead of following one of the HVS models stage by stage, they extract several features from the original and degraded video sequences. Those features were forward statistically analyzed in comparison with the corresponding human rating extracted form subjective tests. This analysis provides the parameters that adjust the objective measures for these features, and after being combined in a simple linear model, they provide the final predicted scores. Some of the extracted features require the presence of the original sequence while others are extracted in a no reference mode. The proposed metric exploits spatial and temporal information. The processing includes Soebel filtering, Laplace filtering, fast Fourier transforms, first-order differencing, color distortion measures, and moment calculation.

- Based on previous works, the ITS in [140] proposed a RR metric for in-service quality monitoring system. Their metric is built on a set of spatio-temporal distortion metrics that can be used for monitoring in-service

of any digital video system. Authors show that a digital video quality metric, in order to be widely applicable must accurately emulate subjective responses, must work over the full range of quality (from very low bit rate to very high), must be computationally efficient, and should work for end-to-end in-service quality monitoring. The metrics presented in their work are based on extracted features from the video sequence as in [170], and in order to satisfy the last condition (to be able to work in in-service monitoring systems), these features, extracted from spatio-temporal regions, are sent, compressed following the ITU-R Recomendation BT.601 through an ancillary data channel so that it can be continuously transmitted. In the paper, the authors describe these spatio-temporal distortion metrics in detail so that they can be implemented by researchers.

- Later, through the National Telecommunications and Information Administration (NTIA), the same authors proposed the General Model of the Video Quality Measurements Techniques (VQM) metric for estimating video quality and its associated calibration techniques. This metric was submitted to be independently evaluated on MPEG-2 and H.263 video systems by the Video Quality Experts Group (VQEG) in their Phase II Full Reference Television (FR-TV) test. The VQM, which is based on the same algorithms used in their previous works [170, 140] was standardized by the VQEG, and a technical report [171] was supplied with a full description of the metric and all its operation modes. This metric was later summarized in [172]. As mentioned before, the VQM uses RR parameters that are extracted from optimally-sized spatio-temporal regions of the video sequence. The ancillary channel and the calibration techniques require at least a 14% of the uncompressed sequence bandwidth. Information is sent through that channel. Although being conceptually a RR metric, it was submitted to the VQEG FR-TV test because the ancillary channel can be used to receive more detailed and complete references from the original frames, even the original frames themselves. The VQM with its associated calibration techniques comprise a complete automated objective video quality measurement system. The calibration techniques include spatial alignment, valid region estimation, gain and level offset calculation, and temporal alignment. Finally, in [173], the authors reduce the requirements of some of the features extracted in the NTIA General Model in order to achieve a monitoring system that uses less than 10 kbits/s of reference information.

- In [174], the authors propose a NR metric for blocking artifacts in images. Previous NR blocking metrics measured the amount of blocking by using a weighted mean-squared difference along block boundaries [175]. This method can produce situations in that even the original image can be

evaluated as blocky. The authors propose to treat the distorted image as a pure non-blocking image that is interfered with a pure blocky signal, and the key of the metric is to measure the power of that blocky signal. They define an ideal 1-D blocky signal that is suppose to interfere the original image for each row and column. For measuring the amount of blocking, they use a power spectrum estimator of the image in the Fourier domain, i.e., after applying the Fast Fourier Transoform (FFT). A final weighting and summing stage that processes row and column information produces the final blocking measure.

- The authors in [176] propose another NR metric for blocking artifacts; this work was extended in [177]. Their metrics work in the DCT domain. They first define a 2-D step function for modeling an overlapping block that is made of the bottom and upper part of vertically adjacent blocks, or left/right for horizontal adjacent blocks. Once they have modeled the 2-D step function of that *overlaping block* and are able to measure the amount of edge activity (blocking) in the DCT domain, they include the luminance masking and the texture masking in the process. Although more accurate models have been proposed in the literature, they propose a simple model of texture masking artifacts to facilitate real-time operations using the amplitude of the 2-D step function, and the amount of blocking measured for the horizontal and vertical edge activity. For luminance masking, they adopt the model proposed in [178]. Finally, they produce a map of *artifact visibility* for the whole image so that block artifacts reducing algorithms can adaptively work according to local visibility. They also provide a combined numerical value as a global blocking artifacts measure in the image.

- A NR metric for blocking and blurring and specifically designed for JPEG compressed images is presented in [179] with low computational cost. The authors provide a Matlab implementation of the metric and the value for their model parameters obtained so that the results can be reproduced. The metric measures blocking and blurring, combining both together to get the final image score. First they calculate for each row a new row that holds the differences with the previous row. This *differences image* is used to calculate next values. The blockiness measure is estimated as the average differences across block boundaries and the blurring is calculated using the activity of the image signal. The activity is calculated using the average for in-block differences and the zero-crossing rate for each block. Zero-crossing occurs when for a *differences row* the difference value for a specific column crosses zero, i.e., previous column has a positive value and next column negative or vice versa. Finally, the blockiness and the two activity measures are modeled in an equation whose parameters are obtained by fitting the MOS values of various test image sets.

- A NR perceptual blur metric is presented in [180] that is based on the analysis of the spread of the edges in an image. They argue, based on a correlation with MOS values, that measuring the spread of vertical edge is sufficient to model the perception of blur, avoiding repeating the measures for horizontal edges or in the direction of the gradient of those edges. They use a Sobel filter to detect vertical edges and measure the local blur for each row as the width of the edge. Averaging this local blur for all the edge locations on the whole image, they get the final blur measure. To detect the width of each edge detected with the Sobel filter, the beginning and end pixels are determined by searching around the edge location the local maximums and minimums for each row. Their proposal has low computational complexity and its performance is independent of the image content.

  In [181], the same authors extended their work to include the aforementioned NR blur metric with a FR Blur metric and a FR Ringing metric. The proposed metrics are defined in the spatial domain with very low complexity and are based on the analysis of the edges in an image. The blur metrics measure the spread of the edges and the ringing metric measures oscillations around edges. In the FR version, the edges used for their algorithm are those from the original image while in the NR version, the edges are obtained directly form the processed or compressed image. The ringing metric is based on the FR blur metric. From the wavelet decomposition filters, they obtain a fixed ring-width. The edge width measured by the blur metric is substracted from that ring-width. The resulting width is the distance around the edge (left and right) where differences (oscillations) with the original image are locally measured for each edge position. The difference between the maximum and minimum difference in the ring-width (left and right) is multiplied by the ring-width itself, giving the amount of ringing for each edge position. Averaging for all edge positions in the image they obtain a global ringing measure. They finally combine both metrics (blur and ringing) to a FR quality metric.

- The Reduced Reference metric called Hybrid Image Qualitiy Metric (HIQM) proposed in [182] is a weighted sum of different image artifact measures (smoothness, blocking, ringing, masking, and lost block/pixel). The blocking measurement is based on the algorithm proposed by Wang et al. [174, 179]. The blur measurement algorithm is based on previous work in [180]. They use the metric proposed in [183] to detect ringing and lost blocks by measuring the edge activity and the gradient activity that is higher in the distorted image due to the apparition of false edges. Finally, masking detection is based on the global contrast measure of the image that is in turn based on the standard deviation of the first-order image histogram that is used to measure the average brightness of the

image. A weight is given for each distortion and an averaged weighted sum produces the final quality value of the metric. The weights are empirically obtained in order to achieve good correlation with PSNR.

- The proposal of [184] includes another way to assess the quality of images. In this case, images to be judged are improved versions of the original ones, i.e., they try to predict the quality of enhanced images. The authors argue that the Error Sensitivity approach or the use of RR or NR metrics that are based on properties of the distorted image are not suitable for this task because those methods are designed to assess the quality of degraded images. So they propose to use a neural network that has been trained to predict the final quality of the enhanced images as it would be judged by human assessors. The inputs to the neural network are numerical values corresponding to several objective properties of the enhanced image. These values are determined at the signal level, i.e., are based on pixel values that are extracted block by block (block size, 32x32 pixels). These features describe the image content in terms of luminance distribution, spatial orientation, frequency energy distribution, etc.

- As in other proposals, the authors in [185] propose the use of a RR metric to assess the quality of a video sequence. They use image properties or indicators to measure differences between the original and distorted image that are encoded and transmitted with the video sequence. So at the decoder side, the same properties are obtained from the distorted image and compared with the original ones. The authors use this RR metric in combination with another NR metric to assess quality of video streaming over IP networks. The RR metric accounts for image quality while the NR metric accounts for transmission quality. The basic indicators for the RR metric include the Estimated Additive Gaussian Noise power level (based on Wiener filtering), the Impulsive Noise power level estimation (based on median filtering), Blocking and Blurring artifacts (based on [174, 179]) and finally, statistics of Ringing Artifacts (based on a Perona-Malik filter). These properties are embedded in the coded bitstream. The NR component mainly refers to the impact of temporal resolution reduction, packet losses, latency, and delay jitter. Although packet loss and out of sequence ratios can be derived by gathering the communication channel output, authors use only the decoded information to detect these effects.

- Other metrics that take advantage of the contrast masking effect of the HVS are included in this framework. So, we can find metrics based on watermarking techniques that analyze the quality degradation of the embedded image [186]. Also, in the metric presented in [187] based on a new concept named *Quality-aware image*, authors extract some features of

the original image that are embedded into the image as invisible hidden messages. When the distorted image is received, the loss of parts of that hidden features yields to an objective measure of the quality of the received image.

- In [188], a Weighted MSE (WMSE) measure is proposed, where local luminance, contrast sensitivity, and masking are taken into account in the proposed quality index. They use a variation of the filter bank decomposition proposed by Simoncelli et al. in [163]. As a result of the filter operation on the image they get three channels, luminance $Y_{ij}(x,y)$, red-green $RG_{ij}(x,y)$, and yellow-blue $JB_{ij}(x,y)$, where i indexes the radial frequency band and j indexes the orientation with center frequency of $j\pi/4$. The decomposition in the frequency domain resembles the Simoncelli decomposition and is shown in Figure 2.30. After the filtering stage, the luminance and chrominance channels are converted into a measure of local contrast for each band and orientation by normalizing each channel with the global average luminance of the image. To include the contrast sensitivity, they calculate a sensitivity function for each channel but using different CSF models for luminance and chrominance channels. For the luminance channels, they use the Barten CSF function [189], and for chrominance they use the Martin, Ahumada and Larimer CSF function [190]. To obtain a Weighting factor for each channel, they apply each of the CSF models to the channel functions for the central frequency in each band. And finally, they apply a threshold elevation function to account for contrast masking. They provide a final WMSE that is the weighted sum of all MSE's in all frequency bands, orientations, and luminance and chrominance components.

2.5.3 Statistics of natural images framework

Some drawbacks of the Model Based HVS framework are reviewed in [81, 191]. Some of these drawbacks are, for example, that the HVS models work appropriately for simple spatial patterns, like pure sine waves; however, when working with natural images, where several patterns coincide in the same image area, then their performance degrades significantly. Another drawback is related to the Minkowsky error pooling, as it is not a good choice for image quality measurement. As the authors show, different error patterns can lead to the same final Minkowsky error. Also, the HVS Model based framework is designed to estimate the threshold at which a stimulus is just barely visible. These subjectively measured threshold values are then used to define error sensitivity measures as the CSF and various masking effects. But

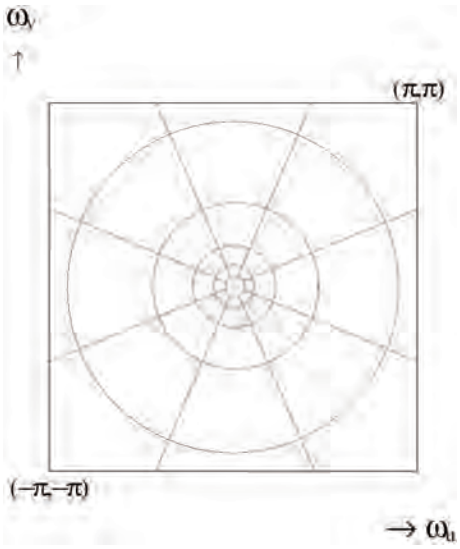


Figure 2.30: Decomposition in the frequency domain used in the WMSE proposal.

most of the impairments produced while processing images are above these thresholds, i.e., are clearly visible, so it is not clear that the near-threshold models can accurately assess suprathreshold distortions. Some studies try to include suprathreshold psychophysics for analyzing image distortions [192, 193, 194].

Therefore, several authors argue that the approach to the problem of perceptual quality measurement must be a top-down approach, analyzing the HVS to emulate it at a higher abstraction level. The authors supporting this approach propose to use the statistics of the natural images. In [195], a review of recent Natural Scenes Statistics (NSS) models is presented.

Some of them propose the use of image statistics to define the structural information of an image. When this structural information is degraded, then the perceptual quality is also degraded. In this sense, a measurement of the structural distortion should be a good approximation to the perceived image distortion. These metrics are able to distinguish distortions that change the image structure from distortions that do not change it, like changes in luminance and contrast.

2.5.3.1 Metrics

- In [81, 196], the authors define a Universal Quality Index (UQI) that is able to determine the structural information of the scene. This index models any distortion as a combination of three different factors: a) the loss of correlation between the original signal and the distorted one; b) the mean distortion that measures how close the mean of the original and distorted version are; and c) the variance distortion that measures how similar the variances of the signals are. The dynamic range of the Quality Index i [-1,1], being 1 the best value, when the signals are identical. They apply this index in a 8x8 window for an image obtaining a quality map of the image. The overall index is the average of the quality map.

- The authors in [191] further improve their previous quality index proposing the Structural SIMilarity (SSIM) (Structural SIMilarity) quality index. This metrics, based on the Universal Quality Index [81, 196] works in the spatial domain. They expose that the index gets better results if it is applied locally and then averaged rather than applying it over the whole image. Applying the SSIM locally reduces the foveation effect, because at typical viewing distances only a part of the image is perceived with high resolution, and can provide a spatially varying quality map of the image. Instead of applying it in a 8x8 block basis as in their previous work, which produces a blocking effect, they use a 11x11 circular-symmetric Gaussian weighting function. They use the Mean SSIM (MSSIM) index to evaluate the overall image quality. Due to the existence of the quality map, the quality of Regions Of Interest (ROI) can be easily computed by averaging the quality in those regions. Several weighting functions can also be applied to the local quality index in order to adapt to any application; however, they use a uniform weighting. This work was later fully explained as a book chapter in [197].

- The authors in [198, 199] proposed a video quality metric following a frame by frame basis. They apply the SSIM index locally in 8x8 blocks randomly selected to reduce computational costs. They apply the SSIM index to the Y, Cb and Cr color components independently and obtaining the global color SSIM index using a weighted summation. Using statistical features like mean and variance, they classify the blocks as smooth region, edge region, or texture region. The results of all the selected areas are averaged to give the frame quality value. This value is further adjusted based on the overall blockiness of the image and the motion factor. The blockiness and blurring are evaluated globally for each frame using the NR metric proposed in [174]. Instead of using a uniform weighting factor while averaging the randomly selected blocks, they assign different weights based on the local luminance; for example, as dark areas attract hardly the

attention of the viewer these areas get a lower weight. The authors also perform a second adjustment based on how the blur distortion is considered depending on the motion in the scene. The motion information is obtained by a simple block-based motion estimation algorithm with full pixel resolution. The final video sequence quality index is the average of the frames quality values. In a still or low motion frame, severe blurring artifacts are very annoying, but in a large motion frame the same amount of blur is perceived as less important because motion blur occurs at the same time. They give different weights according to the type of the frame motion.

- In [200], extended their SSIM to a new Multi-Scale Structural SIMilarity (M-SSIM) model. The new proposed multi-scale analysis runs a low-pass filter to the images (original and distorted versions) and a downsampling process to the filtered images iteratively. Then, at each of the resulting scales, the SSIM index is applied. After M-1 iterations, the Scale M is obtained being the original resolution the Scale 1. At each scale, the contrast comparison and the structure comparison of the SSIM is applied whereas the luminance comparison is applied only at Scale M. The final multi-scale SSIM index is obtained by a weighted combination of the comparison operators. Different weights can be applied to each scale, in the same sense as the CSF applies different weights to each frequency subband, they uniformly weight each scale. They perform a subjective test in order to detect the perceptual importance distortions (in increasing grade) applied at each scale. The results of this subjective test provided the perceptually adjusted weights for each scale. The reason why the authors did not use the CSF for this task is because it is typically measured at visibility thresholds levels and using only simplified stimuli (sinusoids) and the purpose of the new M-SSIM is to compare the quality of complex structured images with distortions above threshold.

- As stated in [201], the main drawback of the spatial domain SSIM algorithm is that it is highly sensitive to translation, scaling, and rotation of the image. So, in this work [201], the authors presented the Complex Wavelet SSIM (CW-SSIM) which extend the SSIM method to the complex wavelet transform domain and make it insensitive to non-structural distortions like zoom, rotations, and translations produced by movements of the acquisition devices. This insensitivity works only if these movements or zooms are smaller than the wavelet filters used.

- In [202], the authors propose a general adaptive linear system framework that is able to decompose the distortion between two images into linear combinations of the constituent distortions. One linear combination corresponds to non-structural distortions like luminance and contrast

changes, gamma distortions and horizontal and vertical translations. It is obtained in a pre-procesing step where the weights for each type of distortion are also computed. The other combination corresponds to structural distortions. A frequency decomposition method, based on the DCT transform matrix, is applied to obtain the structural distortions. With the weighted combination of the two types of combination, a QAM is proposed.

Other authors use also statistics of the scene in a different way. They state that the statistical patterns of natural scenes have modulated the biological system, adapting the different HVS processing layers to these statistics. First, a general model of the natural images statistics is proposed. The modeled statistics are those captured with high quality devices working in the visual spectrum (natural scenes). So, text images, computer generated graphics, animations, drawings, random noise or images and videos captured with non visual stimuli devices like Radar, Sonar, X-ray, etc. are out of the scope of this approach. Then, for a specific image, the perceptual quality is measured taking into account how far its own statistics are from the modeled ones.

- In [203], a statistical model of a wavelet coefficient decomposition is proposed; later, in [204] a RR Image Quality Assesment (RRIQA) is presented. The authors use a model of the statistics of natural images in the wavelet transform domain. They work with the steerable pyramid wavelet transform from [163] and use the Kullback-Leiber Distance (KLD) to measure how different the marginal probability distributions of wavelet coefficients in the reference image and distorted images are. This is used as measure of distortion. They find that several well known types of image distortions produce significant changes in the wavelet coefficient histograms that are detected by the metric. They do not assume any distortion model, so the proposed method is potentially useful for a wide range of distortion types. The marginal probability distribution from the distorted image is obtained directly from the decoded wavelet coefficients, but the marginal distribution from the reference must be transmitted to the receiver as RR data. If the histogram bin size is small then the bandwidth required to transmit the RR features is very demanding, but if the histogram bin size is large then the accuracy of the KLD is reduced. But they send only three parameters as RR data. The cue is that the marginal distribution of the coefficient in an individual wavelet subband can be modeled as a two-parameter Generalized Gaussian Density (GGD) model as they refer. The third parameter is the prediction error between the original distribution and the GGD distribution. So, in the receiver side using the GGD parameters and the error prediction, the marginal distribution of the

reference image can be reconstructed. These parameters are computed and sent for each wavelet frequency subband.

- In [205], the authors propose a NR metric (NRJPEG200) that uses a statistics of natural images model in the wavelet domain [206, 207] in conjunction with information of the distortion model of the JPEG2000 encoder. With both information, they build a simplified model that characterizes images compressed by JPEG2000 as well as uncompressed natural images. The statistical model predicts the wavelet coefficient's magnitude conditioned on a linear prediction of the coefficient. The linear prediction is calculated based on two image dependent estimated thresholds and the relationship of the coefficient with its parent, grandparent, and its neighbors. The quantization of wavelet coefficients produces a reduction of the significant coefficients altering these relationships that are used to predict the quality with no reference of the original image.

- Some metrics defined under this approach take the objective quality assessment as an information loss problem, using techniques related to information theory [204, 85]. In [85], the authors propose to approach the quality assessment problem as an information fidelity problem, where a natural image source communicates with a receiver through a channel. The channel imposes limits on how much information can flow from the source (natural image), through the channel (distortion process) to the receiver (human observer). So they model the input and the output of the channel. The natural image is modeled using Gaussian Scale Mixtures (GSM) that have been reported as very appropriate to model the marginal density functions of the wavelet coefficients and the highly space-variant local statistics of a wavelet transformed natural image [208]. The distortion model is a simple attenuation and additive Gaussian noise model in each subband. Given the source and the distortion the Information Information Fidelity Criterion (IFC) is the mutual information between the source and the distorted image, i.e., the statistical information that is shared. An important feature of the IFC is that it does not involve any parameters associated to display devices, data from psychophysical experiments, viewing configuration, or any stabilizing constants. The IFC is not a distortion metric, but a fidelity criterion, i.e., in ranges from zero (no fidelity) to infinite (perfect fidelity).

2.6 QAM comparison

As previously mentioned, each QAM gets the quality of the image/video using their own and specific scale that depends on its design. Therefore, these raw quality scores cannot be compared directly, even though the range of the values (the scale) is the same. In order to compare fairly the behavior of various metrics for a set of images or sequences, the objective quality index obtained from each metric has to be converted into a common scale.

When reviewing the performance comparisons that the authors made in their QAM proposals, few details are provided about the comparison procedure itself, so it is difficult to replicate these results. In addition, different tests, with the same image set and even with the same subjects, can provide slightly varying results for a set of metrics, but as explained in [209], the results should be in line when tests are correctly done.

In VQEG, subjective tests were repeated by several laboratories and the Pearson correlations between results by different laboratories range from 0.924 to 0.986, with mean of 0.97, confirming that even the best test methodologies cannot fully compensate for the uncertainty related to human factors such as test subjects and the consistency and interpretation of instructions. These results suggest also that slightly less consistent MOS scores are obtained in subjective tests carried out with image databases containing several different types of distortions than that obtained when the database has only a specific type of artifacts.

The authors in [209] reviewed the sources of inaccuracy in each step of the QAM comparing processes, shown at Figure 2.31. Test video sequences or images from a database with known subjective scores (MOS or DMOS) are the input to the QAM. The QAM provides its quality indexes or raw scores. Then, regression analysis is used to find a function that maps the obtained raw scores into subjective quality scores. Finally, a correlation analysis is performed to estimate how accurately the subjective scores are predicted from the objective quality indexes. The set of sequences or images in the database are called the metric *training set* because they are used to fix the regression function.

The sources of inaccuracy in this process may be related to many factors such as the reliability of the subjective reference data, the types and degree of the distortions in the images or videos, the selection of the content that made up the training and testing sets, and even the use and interpretation of the correlation indicators. These sources of inaccuracy can lead to quantitative differences when the same QAM is tested by different authors, even when the tests are correctly done.

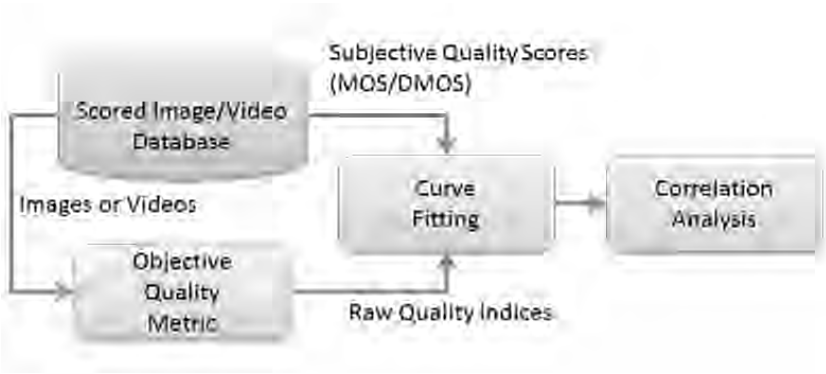


Figure 2.31: Block diagram of the QAM evaluation process

The method in Figure 2.31 is the one proposed by the VQEG [137] with some refinements proposed in other relevant comparison tests [210], where the target scale used is the DMOS scale (Differences Mean Opinion Score). From a a subjective test, for example a Double Stimulus Continuous Quality Scale (DSCQS) method as suggested in [137], the Mean Opinion Score (MOS) can be calculated for the source and distorted versions of each image or sequence in this set. The scale used by the viewers goes from 0 to 100. These scores are converted into difference scores and processed further as explained in [85] to get the DMOS also in the 0-100 range.

The DMOS is the difference between the MOS value obtained for the original image/sequence and the MOS value obtained for the distorted one. So, for a particular image or sequence, its DMOS value provides the mean subjective value of the difference between the original and the distorted versions. A value of 0 means no subjective difference found between the images by all the viewers. Due to the nature of the subjective test, this value is very unlikely.

Performing a subjective test following the recommendations of the VQEG is not an easy and quick task, because a lot of technical requirements must be taken into account and some statistical analysis must be done to the raw subjective data in order to follow VQEG recommendations [136]. So, as shown in Figure 2.31, the source of the subjective scores for such comparison test, is usually an image or video database with the associated MOS or DMOS values.

In [149], the authors review a set of perceptually scored image databases, LIVE [211], CSIQ [212], IVC [213], Toyama [214], A57 [215], TID [216], and WIQ [217]. In addition, some video databases like CSIQ [218], TUM [219], LIVE[93], VQEG-FR-PhaseI [220], and VQEG-HDTV-PhaseI [221] also include subjective values. For the majority of the databases analyzed in [149], the results are in accordance with the results of our tests, which are

shown below.

2.6.1 Metric comparison results

The issues summarized in [209] encouraged and guided us to perform our own comparison test with a set of the most relevant QAM, whose source code or test software has been made available by their authors. The results of our tests, as expected, were slightly different from other comparison tests but remain in line with their results as [209] predicts. The metrics used in our study are summarized herewith.

- The DMOSp-PSNR metric. We translate the traditional PSNR to the DMOS space applying a scale-conversion process. We call the resulting metric DMOSp-PSNR.

- The Mean Structural SIMilarity index [191] (MSSIM) from the Structural Distortion/Similarity Framework. In the reference paper, this FR metric was tested against JPEG and JPEG2000 distortion types. We test its performance with the new distortion types available in the second release of Live Database, *Live2 Database*, since it is considered a generalist metric.

- The Visual Information Fidelity (VIF) metric [222] from the Statistics of Natural Images Framework. A FR metric that quantifies the information available in the reference image, and determines how much of this reference information can be extracted from the distorted image.

- The No-Reference JPEG2000 Quality Assessment (NRJPEG2000) [198] from the Statistics of Natural Images Framework. A NR metric that uses Natural Scene Statistical models in the wavelet domain and uses the Kullback-Leibler distance between the marginal probability distributions of wavelet coefficients of the reference and distorted images as a measure of image distortion.

- Reduced-Reference Image Quality Assessment (RRIQA) [204] from the Statistics of Natural Images Framework. The only RR metric under study. It is based on a Natural Image Statistical model in the wavelet transform domain.

- The No-Reference JPEG Quality Score (NRJPEGQS) [179] from the HVS Properties Framework. A NR metric designed specifically for JPEG compressed images

- The Video Quality Metric[172] (VQM General Model) from the HVS Properties Framework. The VQM uses RR parameters sent through an ancillary channel that requires at least 14% of the uncompressed sequence bandwidth. Although being conceptually a RR metric, it was submitted to the VQEG FR-TV test because the ancillary channel can be used to receive more detailed and complete references from the original frames, even the original frames themselves.

As exposed, the first step in the comparison method is to perform a subjective test to obtain the DMOS values. We have not done such a subjective test. Instead, we have used directly the DMOS values published in the Live Database Release 2 [211] and in the VQEG Phase I Database [220] following the method shown in Figure 2.31. Image metrics were applied to each frame of the sequences and the mean raw value for all the frames was translated to the Predicted DMOS (DMOSp) scale.

As suggested in [209, 223], the performance evaluation of the metrics (Correlation Analysis step) should be computed after a non-linear curve fitting process. A linear mapping function cannot be used because quality scores are rarely scaled uniformly in the DMOS scale because different subjects may interpret vocabulary and intervals of the rating scale differently, depending on the language, viewing instructions, and individual psychological characteristics. Therefore, a linear mapping function would give too pessimistic a view of the metric performance. Several mapping functions could be selected for this purpose, such as cubic, logistic, exponential and power functions, being monotonicity the main property that the function must comply with, at least in the relevant range of values.

The non-linear mapping function between the objective and subjective scores used in our tests was the one suggested by the VQEG and other relevant authors [136, 137, 210], and is shown in Equation 2.4. It is a parametric function that converts the metric raw score into a value in a Predicted DMOS (DMOSp) scale. In this DMOSp scale, the quality score given by a metric for a specific image/sequence is directly comparable with the one given by the other metrics for the same image/sequence.

$$Quality(x) = \beta_1 logistic(\beta_2, (x - \beta_3)) + \beta_4 x + \beta_5 \tag{2.4}$$

$$logistic(\tau, x) = \frac{1}{2} - \frac{1}{1 + \exp(\tau x)} \tag{2.5}$$

Equation 2.4 has five parameters, from β_1 to β_5, that are fixed by the curve fitting process. We have not found in the literature any mapping function jointly with the parameter values for any image/video database. So, we have calculated these parameters based on sets of images and sequences that conforms our *training set*.

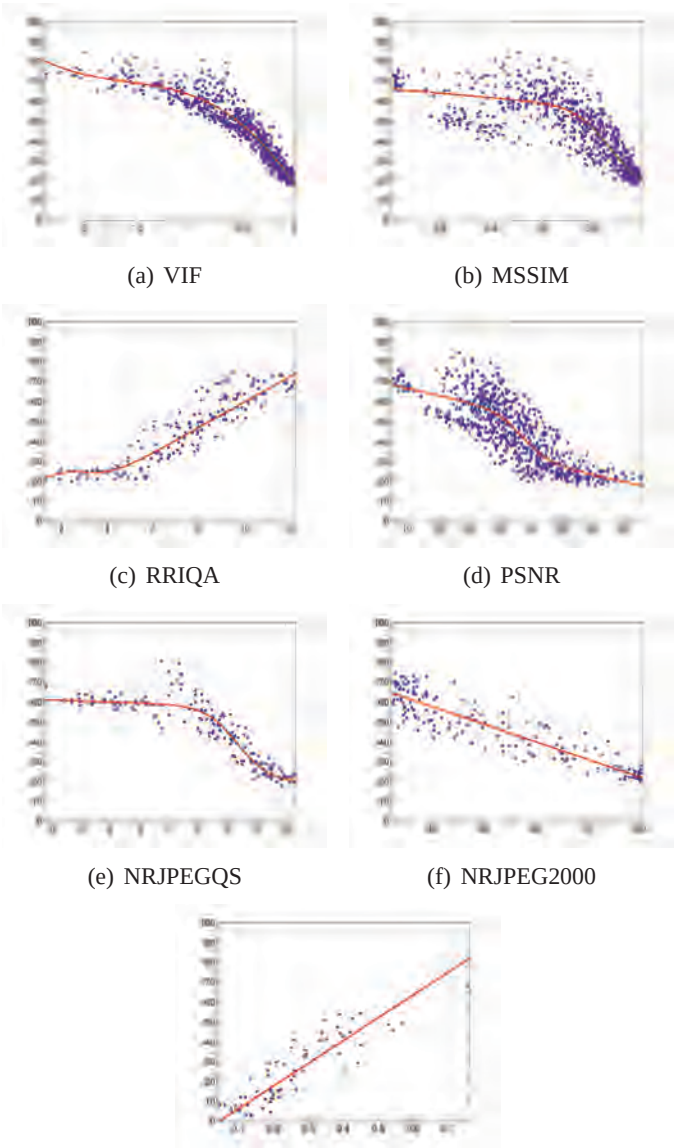
In figures 2.32(a) to 2.32(g), the dispersion plots used in our fitting process for all the selected metrics are shown. Each point of the scatter-plots corresponds to an image in the training set and represents the DMOS value obtained from the scores given by a set of viewers.

The x-axis of the plots correspond to the raw values given by each of the metrics. On the y-axis we have the corresponding DMOS values from the database. The curve fitting process gives us the parameters for Equation 2.4, which is represented by the solid curves. Depending on the metric, increasing x-axis values can have different interpretations, for example, in Figure 2.32(a) for the VIF metric, 0 corresponds to the highest quality reported by the metric and decreasing values mean lower quality, whereas in Figure 2.32(b) for the MSSIM metric, a value of 0 on the x-axis corresponds to lowest quality value being 1 the corresponding value to best reported quality.

The quality of the images in the subjective test is variable, covering a large range of distortion types and intensities for each distortion. Image distortions go from very highly distorted to practically undistorted ones. The viewers gave their scores for each image in the set, obtaining the average DMOS value. As shown in Figure 2.32(a), the dynamic range of the average DMOS values does not reach the limits of the DMOS scale (0 and 100) for any distortion type; therefore, the fitted curve predicts DMOSp values inside the same dynamic range. This is the reason why for a raw score of 0 (the best possible quality for the metric in this case), the predicted DMOSp value is not 0, i.e., there was no image scored with a DMOS value of 0; instead of that, the best DMOSp value obtained is around the value of 20. So, in the case of the VIF metric, its dynamic DMOSp range varies from 20 to 80. The rest of the metrics have slightly different dynamic DMOSp ranges because the set of images used in each case is different, as we explain below.

Once the beta parameters have been obtained for each metric (see Table 2.1), the raw scores can be translated to the DMOSp scale shared by all metrics and hence, we can compare the results given by different metrics while scoring the same image.

The fidelity to subjective scores of a metric is considered high if the Pearson Correlation Coefficient (PCC) and the Spearman Rank Order Correlation Coefficient (SROCC) are close to 1 and the Outlier Ratio (OR) is



(a) VIF (b) MSSIM

(c) RRIQA (d) PSNR

(e) NRJPEGQS (f) NRJPEG2000

(g) VQM

Figure 2.32: Dispersion plots of the evaluated metrics including the curve fit for Eq. 2.4

Table 2.1: Equation parameters of metrics under study

	β_1	β_2	β_3	β_4	β_5
MSSIM	-39.5158	14.9435	0.8684	-10.8913	46.4555
VIF	-3607.3040	-0.5197	-1.6034	-476.0144	-693.3585
NRJPEGQS	37.6531	-0.9171	6.6930	-0.2354	40.7253
NRJPEG2000	37.3923	0.8190	0.6011	-0.8882	74.5031
RRIQA	-18.9995	1.5041	3.0368	6.4301	5.0446
PSNR-DMOSp	23.2897	-0.4282	28.7096	-0.6657	61.5160
VQM-GM	-163.6308	6.3746	-7.6192	114.4685	76.6525

low [148]. In Table 2.2, the performance parameters of our fittings are shown. These performance parameters show the degree of correlation between the DMOSp values and the subjective DMOS values provided by the viewers. Performance validation parameters are the PCC, the Root Mean Squared Error (RMSE), the SROCC, and the OR. In Table 2.3 we include also the Mean, Max, and Standard Deviation (SD) of error. In order to interpret correctly the meaning of *error* is worth to remember that the resulting DMOSp values for each metric correspond to values located on the fitted curve plotted in red in figures 2.32(a) to 2.32(g). So the error for each DMOS point (blue points) is the distance (absolute value) to the fitted curve. Outliers have not been removed from the sets for obtaining these error parameters that provide an idea of how far or close the cloud of points is to the fitted curve in each case.

- The PCC is the linear correlation coefficient between the Prredicted DMOS (DMOSp) and the subjective DMOS. It measures the prediction accuracy of a metric, i.e., the ability to predict the subjective quality ratings with little error.
- The SROCC is the correlation coefficient between the DMOSp and the subjective DMOS. It measures the prediction monotonicity of a metric, i.e., the degree to which the predictions of a metric agree with the relative magnitudes of the subjective quality ratings.
- OR is defined as the percentage of the number of predictions outside the range of 1.5 times the standard deviation of the subjective results. It measures the prediction consistency, i.e., the degree to which the metric maintains the prediction accuracy.
- Mean Error is the mean of the errors produced when obtaining each DMOSp value in relation to their original DMOS value (for all images in the used *training set*).

Table 2.2: Statistical parameters of the goodness of fit

	PCC	RMSE	SROCC	OR
MSSIM	0.8625	8.1809	0.8510	0.0359
VIF	0.9502	5.0187	0.9528	0.0282
NRJPEGQS	0.9360	5.7006	0.9020	0.0455
NRJPEG2000	0.9099	6.7306	0.9021	0.0059
RRIQA	0.9175	6.5393	0.9194	0.0353
PSNR-DMOSp	0.8257	9.0852	0.8197	0.0064
VQM-GM	0.8957	7.6435	0.9021	0.0000

Table 2.3: Error related parameters of the goodness of fit

	Mean Err	Max Err	Std Err
MSSIM	6.2130	24.3351	8.1792
VIF	3.8676	25.4201	5.0219
NRJPEGQS	3.9946	21.9940	5.6562
NRJPEG2000	5.4029	18.4913	6.7506
RRIQA	4.8190	19.2447	6.4961
PSNR-DMOSp	7.2712	24.7603	9.0911
VQM-GM	6.3009	16.4353	7.6897

- Max Error is the highest error produced when obtaining the DMOSp values.
- Std Error is the Standard Deviation of errors

Another key point to consider while comparing QAM [209] is the correct selection of the image or video sequence sets used as *training set*. The *training set* is used to perform the curve fitting process. This set should be chosen with special care and must be excluded from validation tests. So for each metric, the fitting process must be done using images or sequences with impairments that the metric is designed to handle. See [209] for details of how an incorrect selection of the image *training set* can influence the final interpretation of the statistics used in the correlation analysis.

So, the MSSIM, VIF, RRIQA, and DMOSp-PSNR metrics were *trained* with the whole Live2 database because they are intended to be generalist metrics. The NRJPEGQS was *trained* only with the JPEG distorted images of the Live2 database as this metric is designed only to handle these type of distortions. And for the same reason the NRJPEG2000 was *trained* only with the JPEG2000 (JP2K) distorted images of the Live2 database and the VQM-GM was *trained* with a subset of 8 video sequences and its 9

corresponding Hypothetical Reference Circuit (HRC) of the VQEG Phase I database in a bit rate range of 1 to 4Mb/s.

It is important to mention that each of these *training sets* have different dynamic ranges in the DMOS scale as the degree of distortions applied to the images is different in each set.

We define as *homogeneous metrics* those which were trained with the same sets and therefore sharing the same DMOS dynamic range. So, metrics are called to be *heterogeneous metrics* when they were trained with different sets.

In our study, all the metrics have been *trained* only with the luminance information and as suggested, only appropriate impairments are used while conforming the *testing sets*for each metric. .

From the performance results we can conclude than with the images and sequences that comprise our training sets the QAM that best performance gives, i.e. a higher correlation with subjective results, is the VIF metric.

2.6.2 Analyzing metrics behavior

In the next subsections, we are interested in analyzing the metrics behavior when measuring image and video distortions produced in 1) compression scenarios at different bit rates, and 2) distortions produced by packet losses in mobile ad-hoc network scenarios with variable degrees of network congestion and node mobility.

2.6.2.1 In compression environments

In this section, we will study the behavior of the QAM under evaluation when assessing the quality of compressed images and sequences with different encoders. As exposed before, in the development of a new encoder or when performing modifications to existing ones, the performance of the proposals must be evaluated in terms of perceived quality by means of the R/D behavior of each encoder. The distortion metric commonly used in the R/D comparisons is PSNR.

So, in this test environment, we will work with the selected metrics as candidates to replace the PSNR as the quality metric in a R/D comparison of different video codecs. In this case, we will use a set of video encoders and video sequences in order to create distorted sequences Hypothetical Reference Circuit (HRC) at different bit rates, and analyze the results of the different QAM under study. Also, we will consider the metric complexity in order to

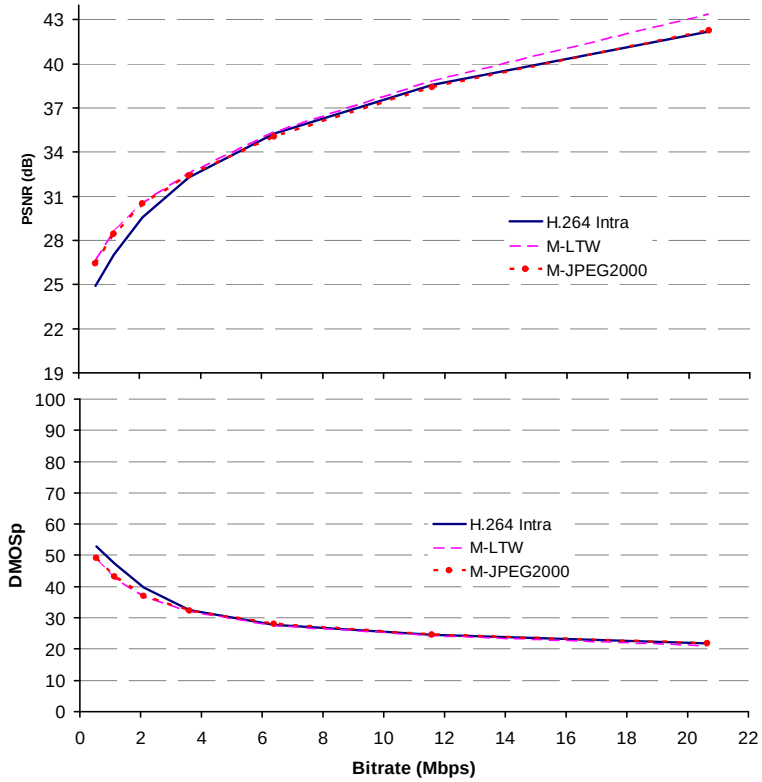


Figure 2.33: PSNR vs. DMOSp-PSNR for the evaluated codecs (mobile sequence)

determine their scope of application. For the tests we have used an Intel Pentium 4 CPU Dual Core 3.00 GHz with 1 Gbyte RAM. The programming environment used is Matlab 6.5 Rel.13. The fitting between objective metric values and subjective DMOS scores was done using the Matlab curve fitting toolbox looking for the best fit in each case. The codecs under test are:

- H.264/AVC [224]
- Motion-JPEG2000 [225]
- Motion-LTW [226]

A R/D plot of the different video codecs under test, using the traditional PSNR as a distortion measure, is shown in the upper panel of Figure 2.33. It is usual to evaluate performance of video codecs in a PSNR range varying from 25-27 dB to 38-40 dB because determining which one is better for PSNR values above 40 dB is difficult.

We convert the traditional PSNR to a metric that we call DMOSp-PSNR by applying the scale-conversion process explained in Section 2.6. We can consider the DMOSp-PSNR metric to be the *subjective* counterpart of the traditional PSNR. It is the same metric, though expressed in a different scale. The DMOSp scale denotes distortion, thereby quality increases as the DMOSp value decreases. The main difference between PSNR and its counterpart, the DMOSp-PSNR, is that the saturation effect is fixed, as we can see in the lower panel in Figure 2.33. As the only modification that has been done to the PSNR metric is the mapping process with the DMOS data, the raw values of the PSNR do not change; therefore, the DMOSp-PSNR metric does not fix the known drawbacks shown in Figure 2.2.

This saturation effect at high qualities is not captured by the traditional PSNR that increases steadily as the bit rate rises, as shown in the upper panel of Figure 2.33. The subjective saturation effect is noticeable above a specific quality value (saturation threshold) where the DMOSp values practically do not change. In our tests the saturation threshold was located at a bit rate of 11.58 Mbps. This behavior is repeated for all the evaluated codecs and video formats, confirming that there is no noticeable subjective difference when watching the sequences at the two highest evaluated bit rates (11.58 and 20.65 Mbps).

For each bit rate value below the saturation threshold, the DMOSp-PSNR metric arranges the codecs (by quality) in the same order as the PSNR does, as expected, because in fact it is the same metric. This quality sorting, below the saturation threshold, agrees also with the results of the subjective tests that we performed (see below), and this behavior is repeated for all the evaluated sequences and bit rates.

Since PSNR, and therefore DMOSp-PSNR, are known to be inaccurate perceptual metrics for image or video quality assessment, we analyze the remaining metrics under study for all codecs and bit rates. From Section 2.6, we know that the expected behavior of a QAM when scoring an image or sequence at different bit rates shoud be:

- For bit rate values below the saturation point, it should provide a decreasing quality value as the bit rate decreases.
- For bit rate values above the saturation point, the perceptual quality value should be almost the same.

So, we ran all the metrics for each HRC (sequence and codec) and analyzed the resulting data between consecutive bit rates, obtaining the quality scores in the DMOSp space. Then, a simple subjective DSCQS test was performed with 23 viewers in order to detect if there were perceptual differences at high bit rates

or not, i.e., above the saturation threshold, for the tested sequences. For each sequence and encoder, the three HRCs with higher bit rates were presented to the viewers, each time in a different order, so that the viewers did not know the rate for each sequence. These HRCs were: the first one located below saturation point (6.4 Mbps) and the two located in the saturation region. For example, in Figure 2.33 these three points are located at 6.4 Mbps (below threshold) and the two rightmost points at 11.58 and 20.65 Mbps. The test shows that:

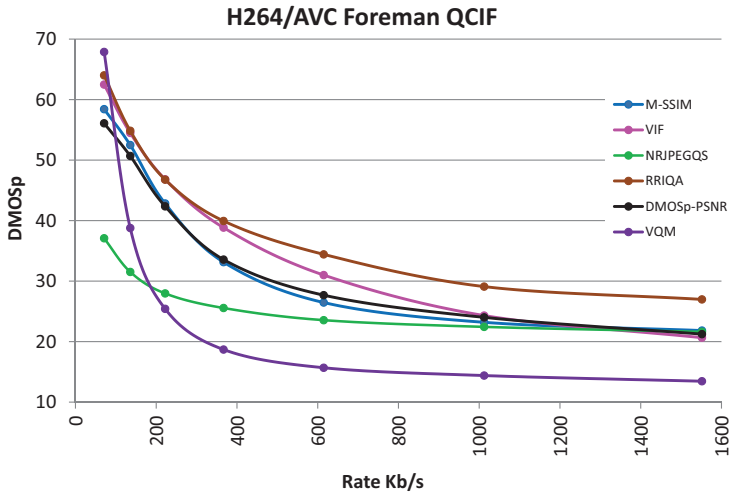
- All the viewers detected some perceptual differences below the threshold.
- No perceptual differences were detected above the saturation threshold.
- Above the saturation threshold, the DMOSp differences for the tested HRCs vary from 0.37 to 6.73 DMOSp points depending on the metric, sequence and encoder. See the whole set of values in tables 2.6 to 2.7 at the end of this chapter.

So, from the results of our subjective test, we can initially conclude that above the saturation differences up to 6.73 DMOSp, values are perceptually indistinguishable.

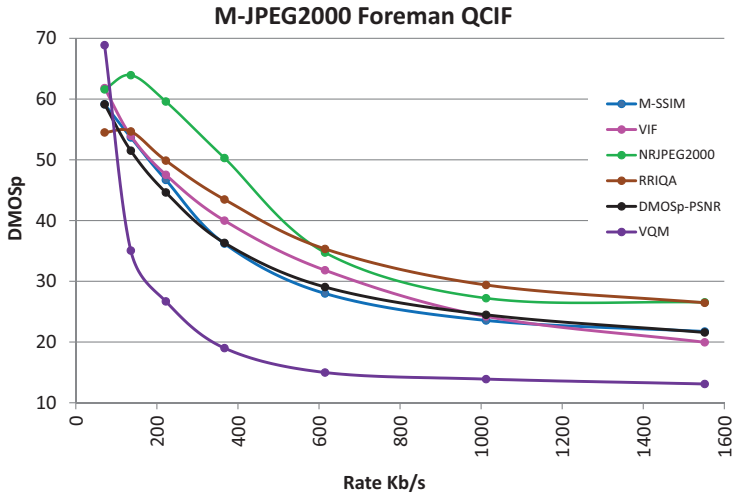
In Figure 2.34, we can see examples of the R/D plots used for comparing the metrics. Each of these figures show the resulting DMOSp R/D curves for all the metrics when applied to the same sequence and encoder at different bit rates. More figures are shown at the end of this chapter in Section 2.8. As shown, in both examples of Figure 2.34, the perceptual saturation effect is captured by all the QAM at high bit rates (high quality) regardless of the encoder. The same holds for all the sequences and encoders.

Some metrics are missing in each of the example plots in Figure 2.34. In the upper plot, the HRCs were encoded with the H.264/AVC codec, and therefore the NRJPEG2000 metric is omitted because it is not designed to handle DCT transform distortions. In the same way, in the bottom plot, where HRCs were encoded with M-JPEG2000, the NRJPEGQS metric is omitted because it is not designed to handle the distortions related to the Wavelet transform.

As mentioned in Section 2.6, monotonicity is expected in the mapping function. So, the expected behavior of the metrics should also be monotonic, i.e. metrics should indicate lower quality values as the bit rates decreases. However, if we look at the lower plot of Figure 2.34, and focus this time on the two lowest bit rates, the quality score given by both, the RRIQA and NRJPEG2000 metrics, increases as the bit rate value decreases. This behavior is contrary to the expected one for a QAM. Remember that lower values of



(a)



(b)

Figure 2.34: QAM comparison using the same sequence with different codecs (a) H264/AVC Intra; (b) M-JPEG2000

Figure 2.35: First frame of Foreman QCIF encoded at 70 Kbps (left) and 135 Kbps (right)

DMOSp represent better perceptual quality. More figures with the same behavior can be found in Section 2.8 at the end of this chapter.

To illustrate this behavior, in Figure 2.35 we show the first frame of the Foreman sequence at these bit rates (for the QCIF frame size). The left image is encoded at 70 Kbps, and the right image at 135 Kbps. After a visual comparison, the right image receives a better subjective score than the left one though the mentioned metrics state just the opposite in this particular case.

Our results for the compression environment stated that:

- NRJPEG2000 offers wrong quality scores between the two highest compression ratios with the M-JPEG2000 codec for QCIF and CIF sequences.
- RRIQA also failed with this NRJPEG2000 at high compression ratios, but only with the QCIF Foreman sequence.
- All the other metrics exhibit monotonic behavior for all bit rates regardless of the encoder and sequence being tested.

Figure 2.34 will also help us illustrate what was exposed in Section 2.6, heterogeneous metrics should not be compared directly, because the dynamic range of the subjective quality scores in each training set is different.

Looking at the upper plot in Figure 2.34 and focusing this time on the lowest bit rate, the DMOSp rating differences between metrics arrive surprisingly up to 30.79 DMOSp units. As the test sequence at this rate is the same for all metrics, this difference seems to be too high and leads us to think that something must be wrong here. In addition, there are three different behaviors or trends in the R/D curves. So, let us analyze that phenomena.

The three different trends in Figure 2.34 correspond to the use of three different training sets. As exposed, VQM-GM was trained with VQEG

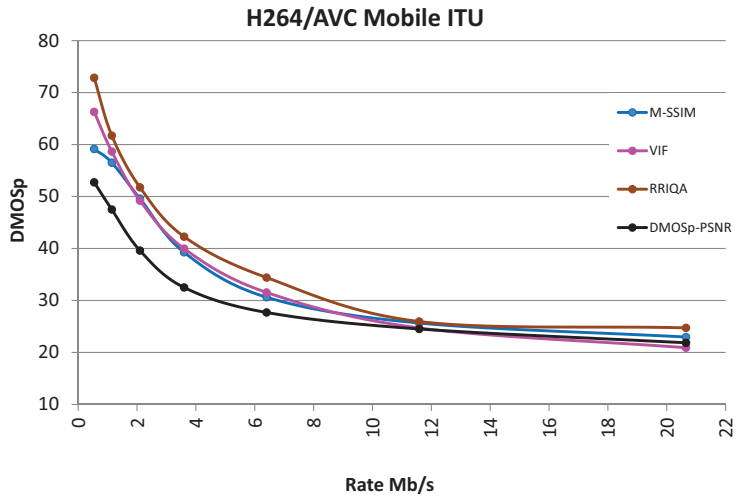


Figure 2.36: QAM comparison plot with homogeneous metrics

sequences, NRJPEGQS was trained only with the JPEG distorted images, and the rest of the metrics trained with the whole set of distorted images in the Live2 database. Each trend is the result of a curve fitting process with different betas (parameters) and these betas are directly dependent on the used training set (the set of distorted images presented to the viewers). This is the reason why the trends and slopes of the metrics below the saturation threshold are different and as shown are *grouped* together in both examples shown in Figure 2.34.

So, when including curves from different metrics in the same R/D plot, it would be preferable that they are homogeneous, and if not, this fact must be told in order to avoid misleading conclusions about the compared performance between heterogeneous metrics. R/D plots with heterogeneous metrics should not be used to determine which metric is the best, not even R/D plots with only homogeneous metrics. These types of plots are useful, however, to analyze the behavior of the metrics for each encoder and/or sequence, to compare and measure differences in quality among metrics while coding at the same rates, and to detect some anomalous behaviors like the ones presented above.

In Figure 2.36, only homogeneous metrics are shown. The trend of all the R/D curves is the same. The best metric can not be concluded only by inspecting the curves and comparing the QAM behavior in the bit rate range. Is it the one with better DMOSp for all the bit rate range? What if this metric is wrongly overrating the quality given by the observers?

Determining how good a metric works at a specific rate or for a bit rate

Table 2.4: Sequences included in the *test set.*

Sequence	Frame	F. Num.	F. Rate
Foreman	QCIF: 176 x 144	300	30 fps.
Container			
Foreman	CIF: 352 x 288		
Container			
Mobile	640 x 512	40	

range depends on how good the metric predicts the subjective scores given by human viewers, i.e., the best metric is the one that best mimics the human rates. This information is obtained from parameters like those of tables 2.2 and 2.3.

Our metric performance validation tests results tells that the VIF metric is the one which best fits the subjective DMOS values among the metrics in the same *training set*. So in plots, such as those from Figure 2.34, the best performing metric can act as reference. Then, we can compare how far from the reference the rest of the metrics are, for each sequence and encoder. Remember that not all the metrics can be used to score all the encoders, they should be able to handle the encoder specific produced distortions.

Once we have compared and analyzed the metrics behavior, and chosen the best correlated one to human perception, we proceed with the encoder comparison. For this comparison, our *test set* comprises different standard video sequences commonly used in video coding evaluation as shown in Table 2.4, using only the luminance component. We perform this test for each evaluated QAM.

Figure 2.37 represents an example of one of the R/D plots used for comparing the performance of the encoders being tested. In this case, the plot shows how the VIF metric evaluates the performance of the encoders. In figures 2.62 to 2.96 the rest of the metrics plots are shown.

For metrics *trained* with the same set, the ranking order of the encoders at a specific bit rate should agree among metrics and also with the subjective ranking given by the viewers. To check this, we performed a simple subjective test with 23 viewers in order to evaluate if we can trust the codec ranking order given by each metric, i.e., at a specific bit rate the metric ranks the encoders by quality in the same perceptual order that subjective one.

For each rate and sequence, the reconstructed sequence of each encoder

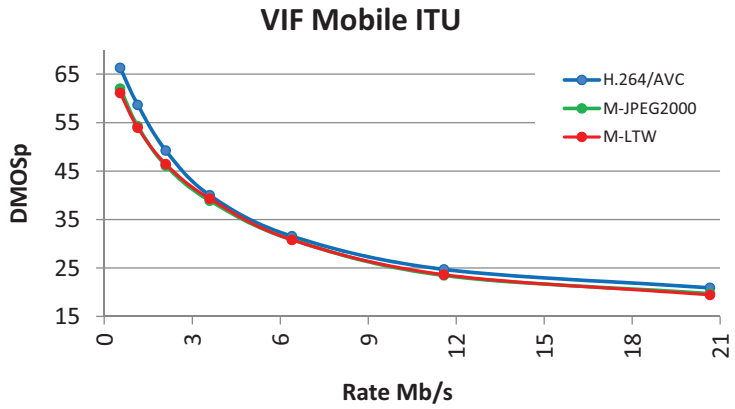


Figure 2.37: R/D performance evaluation of the three video codecs using Mobile ITU video sequence by means of the VIF metric.

was presented simultaneously to the subjects. The ordering of the three sequences varies for each HRC so that the subjects did not know which encoder correspond to each sequence. The subjects ranked the sequences by perceptual quality, and if no differences were detected between pairs of sequences, they annotated this fact. After analyzing the viewer's scores and removing the outliers, the test confirms that the ranking order was consistent among homogeneous metrics, agreeing also with the subjective ranking.

In cases where viewers scored no subjective difference between two sequences, the metrics still gave slightly different values between encoders, and these differences fell in a range lower than 2.9 DMOSp units. When these differences between metric values were higher, for example 3.11 DMOSp units at 2.1 Mb/s between H264/AVC and M-JPEG2000 in Figure 2.37, most of the viewers could see some perceptual differences between the sequences, since they ranked H264/AVC to have better perceptual quality than M-JPEG2000 and Motion LTW (M-LTW).

In order to determine how much difference, expressed in the DMOSp scale, is perceptually detectable, deeper subjective tests and research must be done, because from our studies, we have already detected that the perceptual meaning of these DMOSp differences depend on the point on the DMOSp scale we are working on. For example, for high quality (as stated before), DMOSp value differences up to 6.73 DMOSp points were imperceptible; however, at lower quality levels, smaller differences (3.11 DMOSp points) were perceived.

Finally, Table 2.5 shows, grouped by frame sizes, the mean frame evaluation time and the evaluation time for the whole sequence that each

Table 2.5: QAM Average scoring times (seconds) at frame and sequence level.

	QCIF		CIF		640 x 512	
	Frame	Seq	Frame	Seq	Frame	Seq
MSSIM	0.028	8.4	0.147	44.1	0.764	30.5
VIF	0.347	104.1	1.522	456.5	6.198	247.9
NRJPEGQS	0.01	3	0.049	14.6	0.201	8.1
NRJPEG2000	0.163	48.9	0.486	145.9	1.595	63.8
RRIQA(f.e.)	4.779	1433.7	6.95	2084.9	10.111	404.5
RRIQA(eval.)	0.201	60.2	0.635	190.6	2.535	101.4
DMOSp-PSNR	0.001	0.3	0.006	1.7	0.02	0.8
VQM-GM	0.023	6.975	0.093	27.900	0.300	12.024

metric spent to assess its raw quality value.

In the test, we have disaggregated the time spent on performing the quality comparison from other times spent on performing other steps, for some metrics. This way we can compare times jointly or separately. For example, times spent on the two steps of RRIQA, features extraction (f.e.) and quality evaluation (eval.), have been measured separately.

So, for example if we do not take into account calibration and color conversion times when comparing against the VQM-GM, for CIF sequences the VQM-GM is faster than the other metrics, except NRJPEGQS and DMOSp-PSNR.

DMOSp-PSNR is the least computationally expensive metric for all frame sizes. On the other hand, RRIQA and VIF are the slowest metrics (as they run the Steerable-pyramid; a linear multi-scale, multi-orientation image decomposition).

2.6.2.2 In MANET environments

Our objective in this section is to analyze the behavior of the candidate metrics in the presence of packet losses under different Mobile Ad Hoc Networks (MANET) scenarios. In order to model the packet losses in these error prone scenarios, we use a three-state Hidden Markov Model (HMM) and the methodology presented in [227]. HMMs are well known for their effectiveness in modeling bursty behavior, relatively easy configuration, quick execution times, and general applicability. So, we consider that they fit our purpose of accelerating the evaluation process of QAM for video delivery

applications on MANET scenarios while offering results similar to the ones obtained by means of simulation or real-life testbeds. Basically, by the use of the HMM, we define a packet loss model for MANET that accurately reproduces the packet losses occurring during a video delivery session.

The modeled MANET scenario is composed of 50 nodes moving in an 870x870 square meter area. Node mobility is based on the random way-point model, and speed is fixed at a constant value between 1 to 4 m/s. The routing protocol used is the acDSR protocol Ėvery node is equipped with an Institute of Electrical and Electronics Engineers (IEEE) 802.11g/e enabled interface, transmitting at the maximum rate of 54 Mbit/s up to a range of 250 meters. Notice that a QoS differentiated service is provided by IEEE 802.11e [228]. Concerning traffic, we have six sources of background traffic transmitting File Transfer Protocol (FTP)/Transmission Control Protocol (TCP) traffic in the Best Effort **MAC!** (**MAC!**) Access Category. The foreground traffic is composed by real traces of an H.264 video encoded (using the Foreman CIF video test sequence) at a target rate of 1 Mbit/s. The video source is mapped to the Video MAC Access Category.

We apply the HMM described above to extract packet arrival/loss patterns for the simulation traces, and later replicate these patterns for testing. We describe two environments: (a) a congestion related environment, and (b) a mobility related environment.

The congestion environment is composed of 6 scenarios with increasing levels of congestion, from 1 to 6 video sources. The mobility environment is composed of 3 scenarios with only one video source, but with increasing degrees of node mobility (from 1 to 4 m/s).

For each of these scenarios, we get different packet loss patterns provided by the HMM that represents each scenario.

After an analysis of the packet losses, different patterns are defined:

- Isolated small bursts represent less than 7 consecutive lost packets. As each frame is split in 7 packets at the source, isolated bursts will affect 1 or 2 frames, but none of them will be completely lost. This error pattern is mainly due to network congestion scenarios where some packets are discarded due to transitory high occupancy in the wireless channel or buffers at relaying nodes.

- Large packet loss bursts. Large Bursts cause the loss of one or more consecutive frames. Large packet error bursts are typically a consequence of high mobility scenarios where the route to the destination node is lost and a new route discovery process should be started. This will keep the

network link in down state during several seconds, losing a large number of consecutive packets.

We have used the H.264/AVC codec, adjusting the error resilience parameters to the values proposed in [229] so that the decoder is able to reconstruct sequences even when large packet loss bursts occurs. H.264/AVC is configured to produce one I frame every 29 P frames, with no B frames, and to split each frame in 7 slices, so we put each slice into a separate packet and encapsulate its output in Real-time Transport Protocol (RTP) packets. As suggested in [229], we also force 1/3 of the macroblocks of each frame to be randomly encoded in intra mode.

We have used the Foreman CIF seq. (300 frames at 30 fps) to build an extended video sequence by repeating the original one up to the desired video length. After running the encoder for each extended video sequence, we get RTP packet streams. We will apply a packet erasure process to them, removing those packets declared lost by the HMM model. This process simulates packet losses in the MANET scenarios, so a distorted bitstream will be delivered to the decoder. The decoder behavior depends on the packet loss burst type as follows:

- When isolated small bursts appear, the decoder is able to apply error concealment mechanisms to repair the affected frames. The video quality decreases, and just after the burst, the reconstructed video quality recovers the quality by means of the random intra-coded macroblock updating. When the next I frame arrives, it completely stops error propagation.
- When the decoder faces large bursts, it stops decoding and waits until new packets arrive. This produces a sequence in the decoder that is shorter than the original one. Therefore, both sequences are not directly comparable with the QAM and so we freeze the last completely decoded frame until the burst ends.

Once we have comparable video sequences (original and decoded video sequences with the same length), we are able to run the QAM. Each metric produces an objective quality value for each frame in its own scale. Then, we perform the scale conversion to the DMOSp scale (see Section 2.6).

Figure 2.38 shows the objective quality value in the traditional PSNR scale at three different compression levels (Low compression, Medium compression and High compression) during a large packet loss burst. We observe the evolution of quality during the burst period. What the observer sees during this large burst is a frozen frame with more or less quality, depending on the

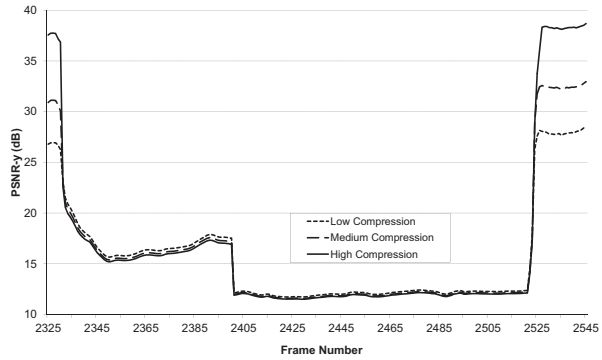


Figure 2.38: PSNR frame values during a long packet loss burst (from frame 2327 to 2525) at different bit rates.

compression level. The PSNR metric reports that quality drops drastically with the first frame affected by the burst, and decreases even more as the difference between the frozen frame and the current frame increases. An additional drop of quality can be observed nearly at the middle of the burst. It corresponds to a scene change (with the beginning of a new cycle of the foreman video sequence). At this point, the drastic scene change makes the differences between sequences even higher, and the PSNR metric scores with even worse values, reaching values as low as 10-12 dBs.

On the other hand, the perceived quality changes at these levels is quite difficult to evaluate. So, a better perceptually designed QAM should not score such a quality drop in this situation because quality saturates. When the burst ends, quality rapidly increases because of the arrival of packets belonging to the same frame number than the current one in the original sequence (frame 2525 in Figure 2.38).

If during such a burst a QAM takes into account only the quality of the frozen frame, disregarding the differences with the original one (which changes over time), the effect of the burst would remain unnoticed for that metric, i.e., quality remains constant.

Figure 2.39 shows the evolution of the candidate QAM during a large burst (similar to Figure 2.38 but in this case in the DMOSp space). There is a panel for each compression level: the upper panel corresponds to high compression, the central panel to medium compression and the bottom panel to low compression. We observe some interesting behavior that we proceed to analyze.

From a perceptual point of view, quality must drop to a minimum when one or more frames are lost completely and should remain that way until the data

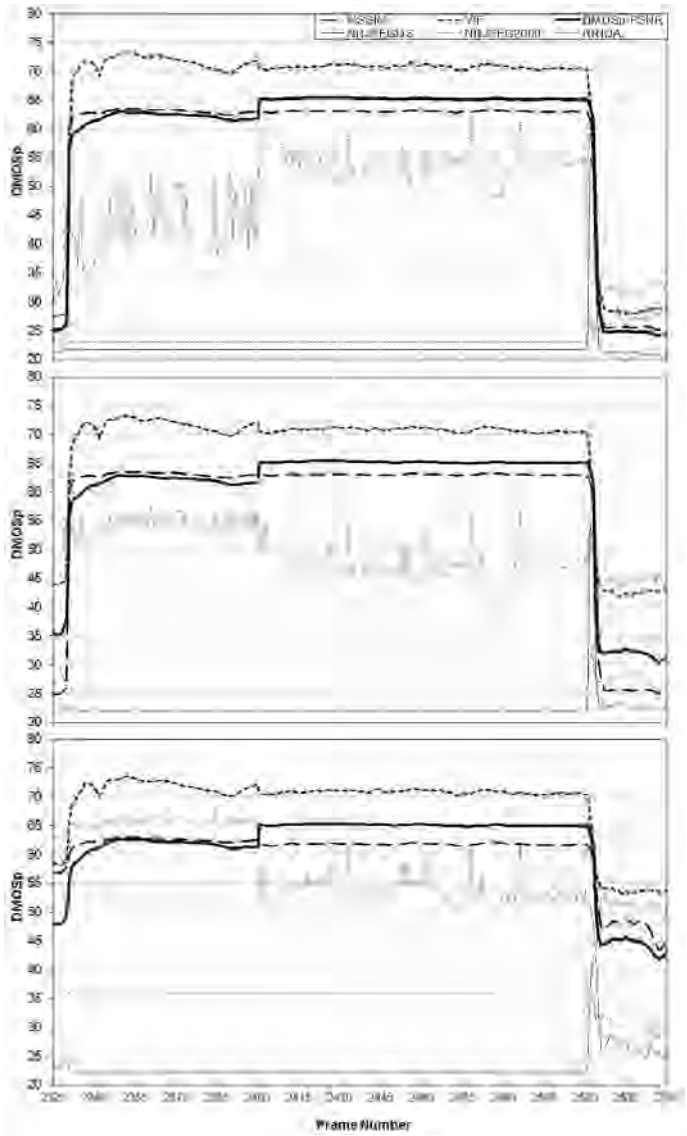


Figure 2.39: Metric comparison in the DMOSp space during a very large burst

(a) (b) (c) (d)

Figure 2.40: Frame reconstruction after a large burst: (a) Original frame, (b) Last frozen frame, (c) and (d) First and second reconstructed frames after the burst.

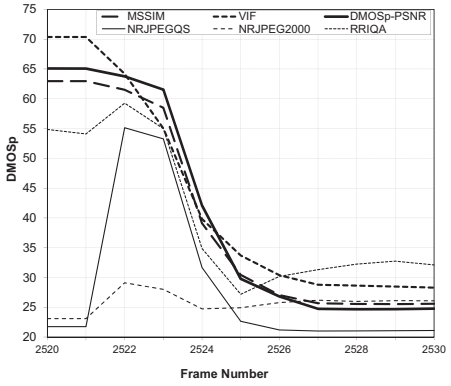


Figure 2.41: End of the large burst for the low compression panel. FR and NR metrics show the opposite behavior.

flow is recovered. It should not matter if a scene change takes place inside the large burst. VIF and MSSIM behave this way. At the point of the burst where the scene change takes place, both the VIF and MSSIM metrics have almost reached their 'bad quality' threshold regardless of the compression level and therefore there is no substantial change in the reported quality. The drop in quality to the minimum at the beginning of the burst provides evidence of the lost of whole frames.

NR metrics do not detect the presence of a frozen frame (by dropping the quality score) as expected because the quality given by these metrics remain at the level scored for the frozen frame during the burst duration. So, NR metrics could not detect the beginning of a large burst, since lost frames will be replaced with the last correctly decoded frame (frozen frame) and the reference frames are not available for comparison. However, NR metrics detect the end of such bursts. Figure 2.40 will help us to explain this behavior, showing how reconstruction is done after a large burst. This figure shows the

impairments produced when the large burst ends. Figure 2.40(a) is the current frame, the one being transmitted. Figure 2.40(b) is the frozen frame that was repeated during the burst duration. When the burst ends, the decoder progressively reconstruct the sequence using the intra macroblocks from the incoming video packets. So, the decoder partially updates the frozen frame with the incoming intra macroblocks. This is shown in figures 2.40(c) and 2.40(d) where the face of the foreman appears gradually.

The gradual reconstruction of the frame with the incoming macroblocks is interpreted in a different way by NR metrics and FR metrics. When the macroblocks begin to arrive, what happens at frame 2522 (see figure 2.41) the NR metrics react scoring down quality, while the FR metrics begin to increase their quality score, just the opposite behavior. For a NR metric, without a reference frame, Figure 2.40(c) has clearly worse quality than Figure 2.40(b). But for a FR metric, the corresponding macroblocks between Figure 2.40(c) and Figure 2.40(a) help to increase the scored quality.

So, NR metrics react only when the burst of lost packets affects frames partially, i.e., isolated bursts, and at the end of a large burst. The NRJPEGQS metric reacts harder (i.e., it shows higher quality differences) than the NRJPEG2000 because it was designed to detect the blockiness introduced by the discrete cosine transform. When the frame is fully reconstructed, then the score obtained with NR and FR metrics again approaches the values achieved before the burst, which depends on the compression rate.

The RRIQA metric shows high variability in its scores between consecutive frames inside bursts. These variations become more evident as the degree of compression decreases. The nature of the data sent through the ancillary channel, 18 scalar parameters obtained form the histogram of the wavelet subbands of the reference image, is very sensitive to loss of synchronism between the reference frame and the frozen one. On the decoder, the same extracted parameters are statistically compared with that received through the ancillary channel. When this comparison is performed with two sets of parameters obtained from different frames, unexpected results appear.

Concerning the FR metrics, MSSIM, VIF, and PSNR-DMOSp show a similar behavior or trend. MSSIM and PSNR-DMOSp show closer quality scores between them than the ones obtained with the VIF metric, which gives lower quality values than the other two metrics. This behavior is the same regardless of the compression level inside the large burst. Leaving aside the PSNR-DMOSp, which is not really a QAM, the other two FR metrics (VIF and MSSIM) have the same behavior when facing large bursts.

Figure 2.42 shows an isolated burst. In this case, blur and edge shifting

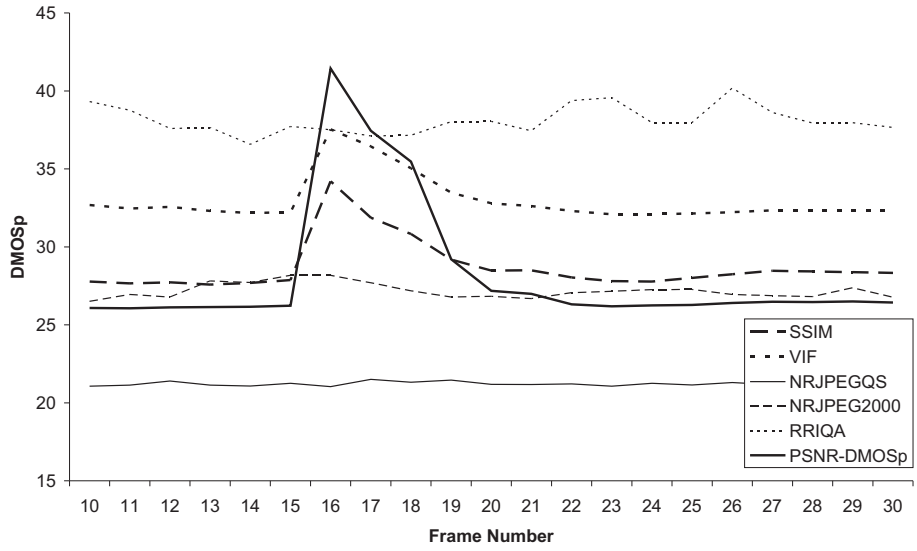
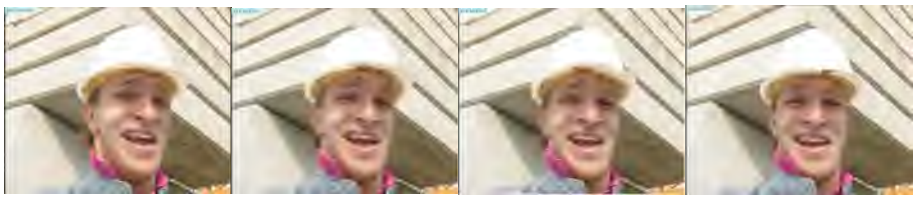


Figure 2.42: Metric comparison for an isolated burst



(a) (b) (c) (d)

Figure 2.43: Packet loss affecting only one frame. (a) Original frame; (b, c, and d) Next three decoded frames.

impairments are introduced altering only one frame. This fact is perceived only by the FR metrics and the NRJPEG2000, which is designed to detect this type of impairment. The error concealment mechanism of H.264/AVC needs up to 6 frames to achieve the same quality scores obtained before the burst. Figure 2.43 shows the original frame (a) and three subsequent frames (b, c, d), where the effect of the lost packets is concealed by the H.264/AVC decoder.

As defined previously, an isolated burst can affect one or two consecutive frames. In the latter case, the behavior of the QAM when facing the isolated burst resembles the behavior of the metrics with a large burst. The difference is that the concealment mechanisms and the correct reception of part of the frames avoid a larger drop in the quality.

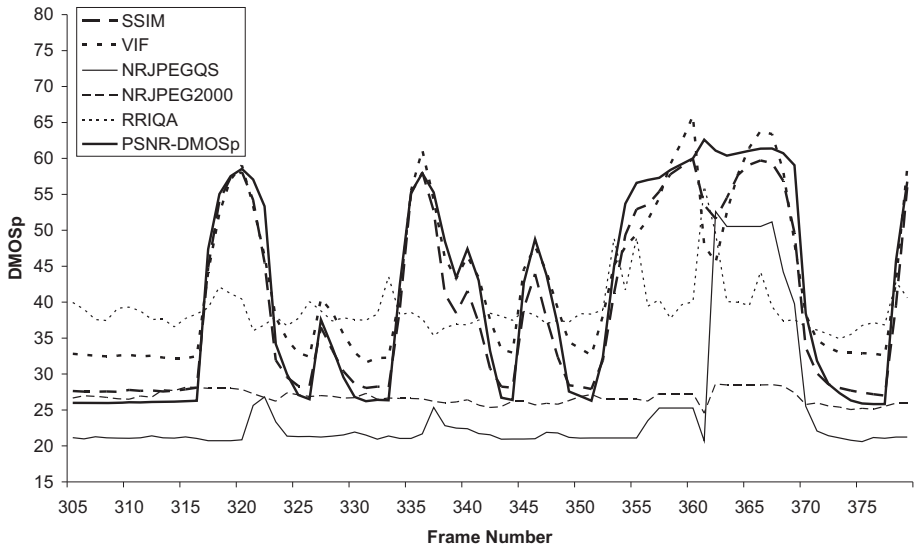


Figure 2.44: Frame interval where different types of bursts occurs consecutively.

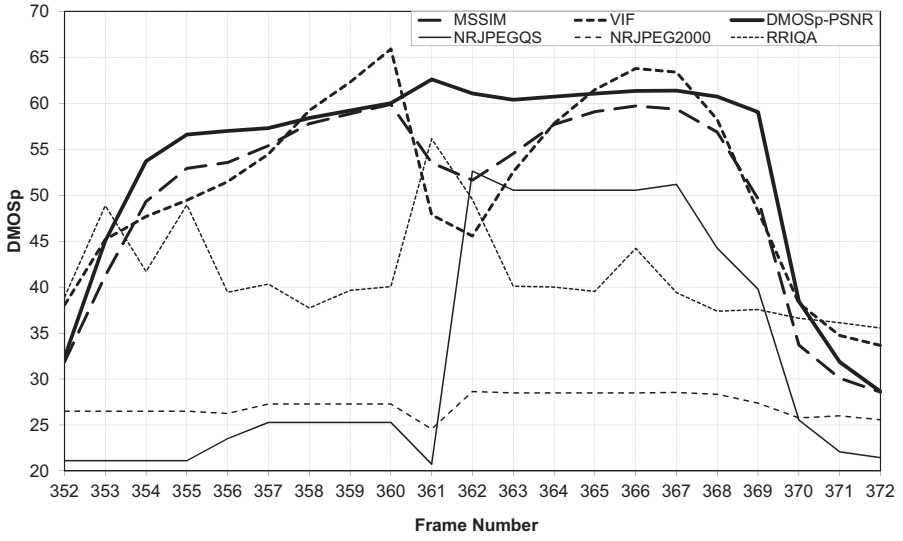
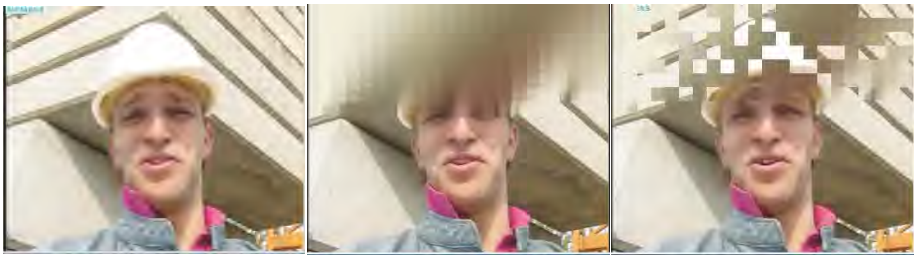


Figure 2.45: Detail from two consecutive long bursts with incoming packets between them.



(a) (b) (c)

Figure 2.46: Decoded frames between two consecutive bursts: (a) original frame; Reconstructed frames: (b) 361 and (c) 362

Figure 2.44 shows multiple consecutive bursts (large and isolated) that behave as exposed previously. From left to right, we see a large burst followed by an isolated one. This pattern repeats again one more time, and at the right most part of the figure, between frames 352 and 372, two large bursts occurs consecutively, having a gap between them where new incoming packets arrive for a short period of time (frames 361 and 362).

In Figure 2.45, we zoom into this area (frames 352 to 372) to analyze why the behavior of the DMOSp-PSNR metric differs from the other FR metrics during the gap between bursts. In the gap, the encoder is not able to reconstruct a whole frame because the gap is too small, i.e., between the two large bursts only a small amount of packets arrive, and this is not enough to reconstruct a whole frame. So the involved frames (361 and 362) are partially reconstructed (figures 2.46(b) and 2.46 (c)). Both frames exhibit perfect correspondence in the lower half with the original one (Figure 2.46(a)). Therefore, the scored quality must increase, at least to some extent, compared to the quality of the previous frozen frame, as occurs at the end of a large burst. This fact is only reflected by the VIF and MSSIM metrics. The PSNR-DMOSp metric is not able to detect this because it is computed using information from the whole frame. For the VIF and the MSSIM, which are perceptually driven, the lower half of the frame increases their raw scores, in the same way as the human scores do. After frame 362, quality decreases again since the following frame is frozen too. So, VIF and MSSIM detect two consecutive loss bursts while PSNR-DMOSp and the other metrics considers only a single larger one.

2.7 Conclusions

The main goal of this work was focused on looking for a Quality Assessment Metric that could be used instead of the PSNR when evaluating compressed video sequences with different encoder proposals at different bit rates, and to analyze the behavior of such metrics when compressed video is transmitted over error prone networks such as MANETs.

We explained the procedures that we followed to compare QAM metrics and alerted about some issues that arise when a comparison between heterogeneous metrics is made. The metrics must be compared using a common scale since the raw scores of the metrics are not directly comparable. The scale conversion process involves subjective tests and the use of mapping functions between the subjective MOS values and the metrics raw values. The parameters for the mapping function we used are provided. The metrics were first trained with a set of images from two open source images and video databases with known MOS values. The metrics were tested with another set of images and videos also taken from available databases. In order to perform a fair comparison, the training and testing sets used with each metric must use only impairments that the metric was designed to handle. We defined as heterogeneous metrics those that were trained with different sets of images or sequences. The R/D comparisons of heterogeneous metrics must be made carefully, focusing not only on the absolute quality scores, but also on the relative scoring between consecutive bit rates as the differences between DMOSp values are perceptually detected (or not) depending on the quality range. When metrics are trained with the same training set, differences in DMOSp values have the same perceptual meaning for all the metrics, but this may not be true between heterogeneous metrics. Normalizing the DMOSp scale when comparing heterogeneous metrics helps to detect these differences.

We performed the comparison between metrics in two environments: a compression environment and a packet loss environment. We performed several subjective tests in order to confirm that the analysis and the behavior of the metrics were consistent with human perception. Our tests included the comparisons of three encoders by replacing the PSNR as distortion metric in their R/D curves with each of the candidate metrics.

From our results in the compression environment, we conclude that we can trust the quality provided by the VIF metric, which is the one that obtains a better fit in terms of DMOS during the calibration process, and also on how it ranks the performance of the tested encoders in the bit rate range under consideration. In the evaluation of the M-JPEG200 encoder, the NRJPEG2000, and RRIQA metrics break monotonicity at very high

compression levels. For the rest of the bit rates, all the other metrics show a monotonic behavior for the entire bit rate range and for all encoders.

In the compression environment with no packet losses, the selection of a QAM to replace the traditional PSNR, depends on the availability of the reference sequence:

- In applications where the reference sequence is not available, RRIQA is our choice because behaves similarly to FR metrics.
- If the reference sequence is available, then choice depends on the weight given to the trade-off between computational cost and accuracy.
  - If time is the most important parameter, we choose DMOSp-PSNR followed by VQM and MSSIM.
  - If accuracy is more important, then the choice will be VIF and MSSIM.

In the loss-prone environment, we analyzed the metrics behavior when measuring reconstructed video sequences encoded and delivered through error prone wireless networks, like MANETs. In order to obtain an accurate representation of delivery errors in MANETs, we adopted an HMM model able to represent different MANET scenarios.

The results of our analysis are the following:

- NR metrics are not able to properly detect and measure the sharp quality drop due to the loss of several consecutive frames.
- The RR metric has a non-deterministic behavior in the presence of packet losses, having difficulties to identify and measure this effect at moderate to high compression rates.
- Concerning the other metrics, MSSIM, DMOSp-PSNR and VIF show a similar behavior in all cases. In summary, we consider that, although they exhibit slight differences in the Packet Loss framework, we propose the use of the MSSIM metric as a trade-off between a high quality measurement process (resembling human visual perception) and computational cost.

2.8 Figures and tables

Table 2.6: Variation in DMOSp values between QAM above saturation point for the Foreman QCIF sequence

	H.264/AVC	M-JPEG2000	M-LTW	Max	Min
M-SSIM	1.36	1.82	1.79	1.82	1.36
VIF	3.65	4.26	4.13	4.26	3.65
NRJPEGQS	0.82			0.82	0.82
NRJPEG2000		0.68	1.21	1.21	0.68
RRIQA	2.12	2.93	2.31	2.93	2.12
DMOSp-PSNR	2.77	2.91	3.34	3.34	2.77
VQM	0.94	0.80	0.82	0.94	0.80
				4.26	0.68

Table 2.7: Maximun and minimun variation in DMOSp values between QAM above saturation point for all the sequences

	Max	Min
Foreman qcif	4.26	0.68
Foreman cif	4.91	0.37
Container qcif	5.88	0.39
Container cif	6.73	0.44
Mobile itu	4.18	0.71
	6.73	0.37

Table 2.8: Variation in DMOSp values between QAM above saturation point for the Foreman CIF sequence

	H.264/AVC	M-JPEG2000	M-LTW	Max	Min
M-SSIM	1.84	2.38	3.32	3.32	1.84
VIF	4.18	3.96	4.91	4.91	3.96
NRJPEGQS	0.87			0.87	0.87
NRJPEG2000		0.82	2.43	2.43	0.82
RRIQA	2.72	2.93	2.03	2.93	2.03
DMOSp-PSNR	2.59	2.52	3.68	3.68	2.52
VQM	0.60	0.37	0.40	0.60	0.37
				4.91	0.37

Table 2.9: Variation in DMOSp values between QAM above saturation point for the Container QCIF sequence

	H.264/AVC	M-JPEG2000	M-LTW	Max	Min
M-SSIM	2.56	2.30	2.30	2.56	2.30
VIF	4.15	4.61	5.06	5.06	4.15
NRJPEGQS	0.90			0.90	0.90
NRJPEG2000		0.45	0.39	0.45	0.39
RRIQA	5.88	4.38	4.04	5.88	4.04
DMOSp-PSNR	2.61	2.66	3.02	3.02	2.61
VQM	1.96	1.88	0.45	1.96	0.45
				5.88	0.39

Table 2.10: Variation in DMOSp values between QAM above saturation point for the Container CIF sequence

	H.264/AVC	M-JPEG2000	M-LTW	Max	Min
M-SSIM	2.47	2.50	2.66	2.66	2.47
VIF	5.07	5.41	5.73	5.73	5.07
NRJPEGQS	0.88			0.88	0.88
NRJPEG2000		0.44	0.48	0.48	0.44
RRIQA	6.73	2.53	1.63	6.73	1.63
DMOSp-PSNR	2.67	2.49	2.90	2.90	2.49
VQM	1.06	0.69	1.14	1.14	0.69
				6.73	0.44

Table 2.11: Variation in DMOSp values between QAM above saturation point for the Moblie ITU sequence

	H.264/AVC	M-JPEG2000	M-LTW	Max	Min
M-SSIM	2.69	3.13	3.10	3.13	2.69
VIF	3.80	3.74	4.18	4.18	3.74
NRJPEGQS	1.45			1.45	1.45
NRJPEG2000		3.62	1.76	3.62	1.76
RRIQA	1.21	2.60	3.85	3.85	1.21
DMOSp-PSNR	2.66	2.84	3.28	3.28	2.66
VQM	0.71	0.81	1.20	1.20	0.71
				4.18	0.71

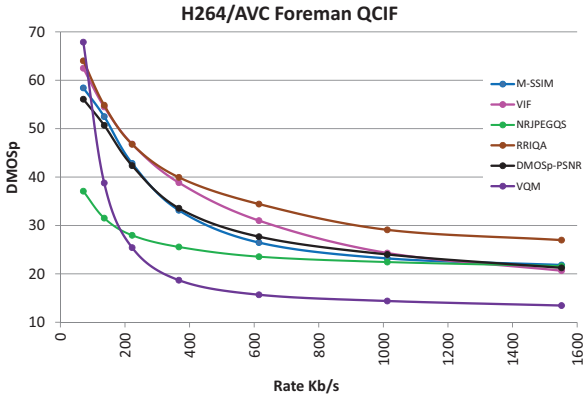


Figure 2.47: QAM comparison for Foreman QCIF and H264/AVC codec in Intra mode.

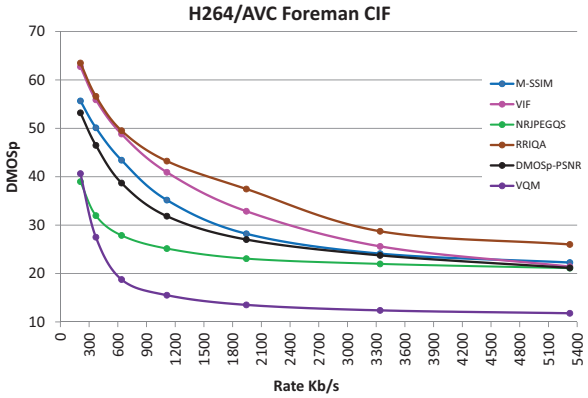


Figure 2.48: QAM comparison for Foreman CIF and H264/AVC codec in Intra mode.

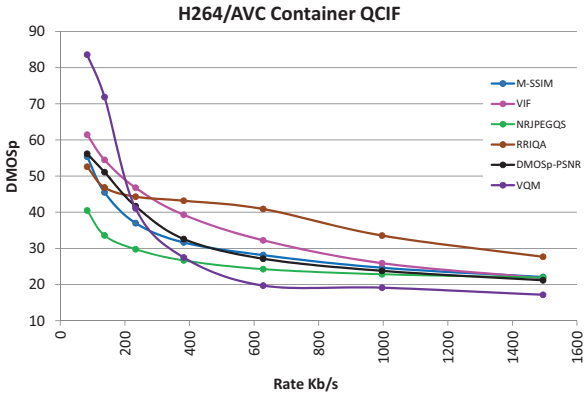


Figure 2.49: QAM comparison for Container QCIF and H264/AVC codec in Intra mode.

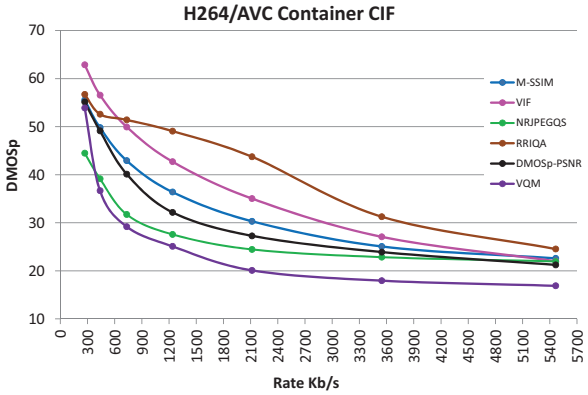


Figure 2.50: QAM comparison for Container QCIF and H264/AVC codec in Intra mode.

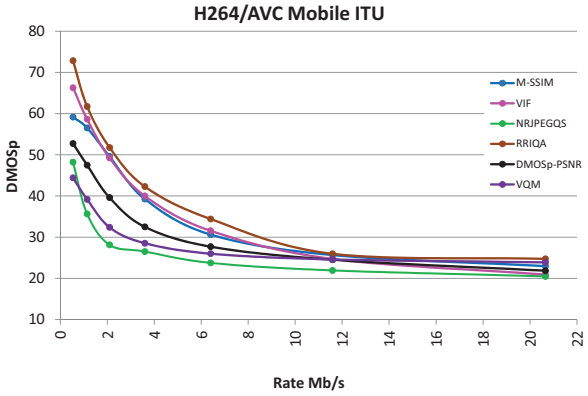


Figure 2.51: QAM comparison for Mobile ITU and H264/AVC codec in Intra mod.e

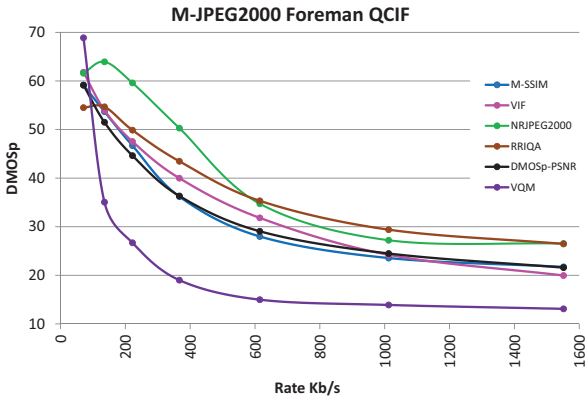


Figure 2.52: QAM comparison for Foreman QCIF and JPEG2000 codec.

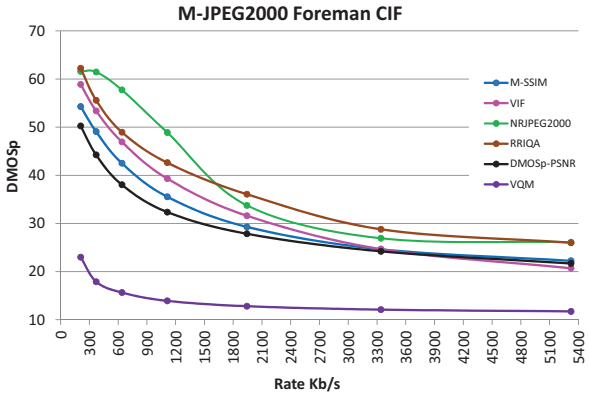


Figure 2.53: QAM comparison for Foreman CIF and JPEG2000 codec.

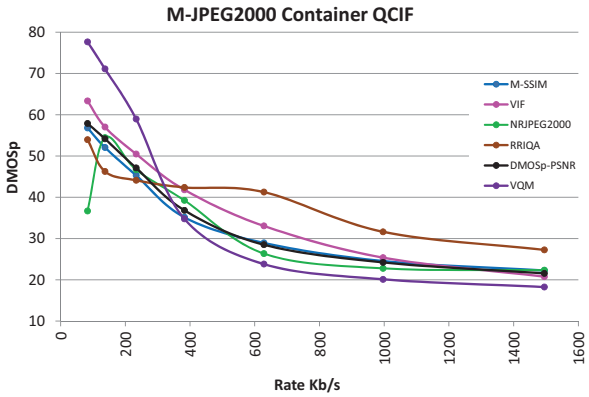


Figure 2.54: QAM comparison for Container QCIF and JPEG2000 codec.

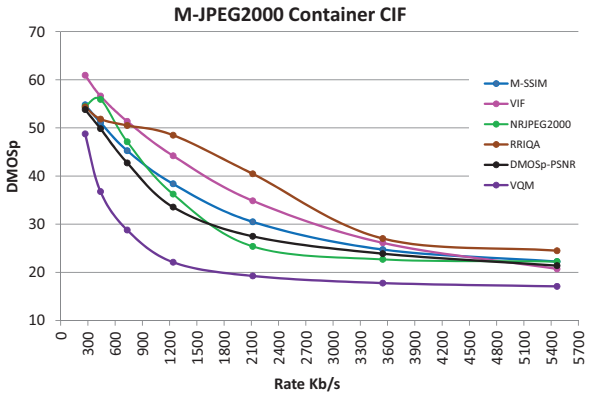


Figure 2.55: QAM comparison for Container CIF and JPEG2000 codec.

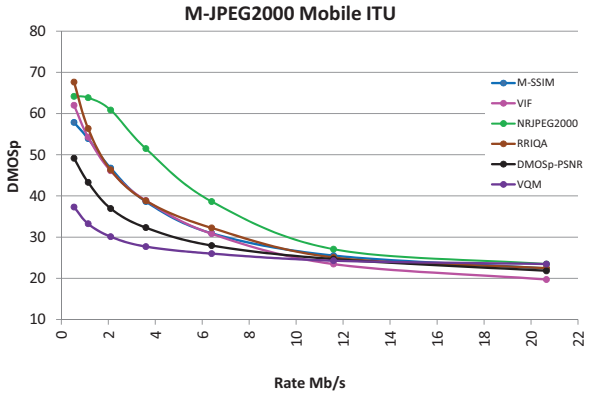


Figure 2.56: QAM comparison for Mobile ITU and JPEG2000 codec.

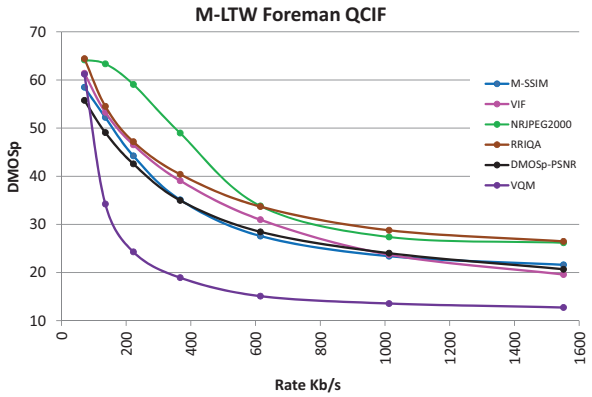


Figure 2.57: QAM comparison for Foreman QCIF and M-LTW codec.

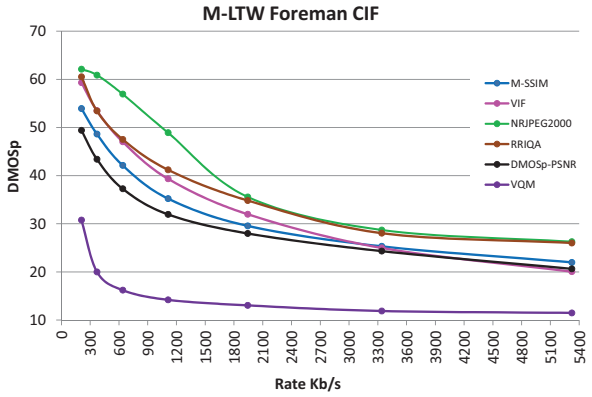


Figure 2.58: QAM comparison for Foreman CIF and M-LTW codec.

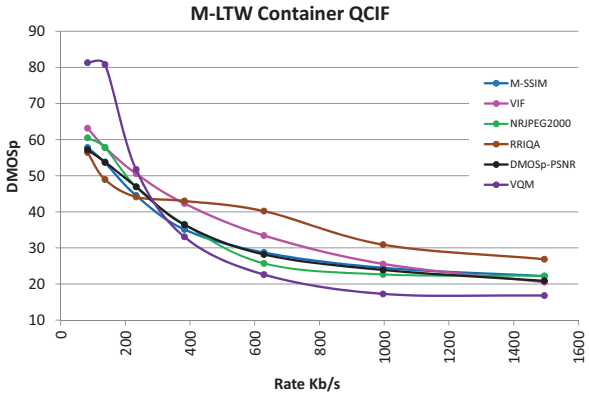


Figure 2.59: QAM comparison for Container QCIF and M-LTW codec.

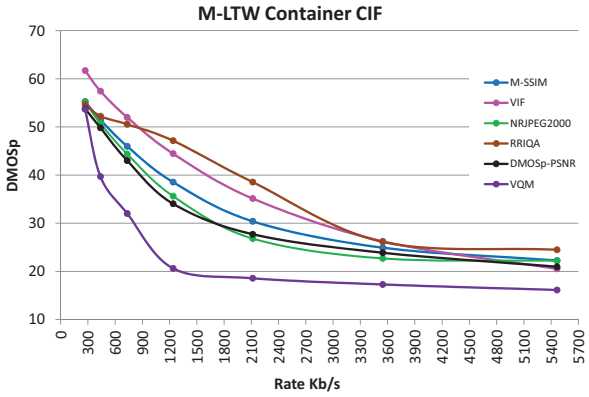


Figure 2.60: QAM comparison for Container CIF and M-LTW codec.

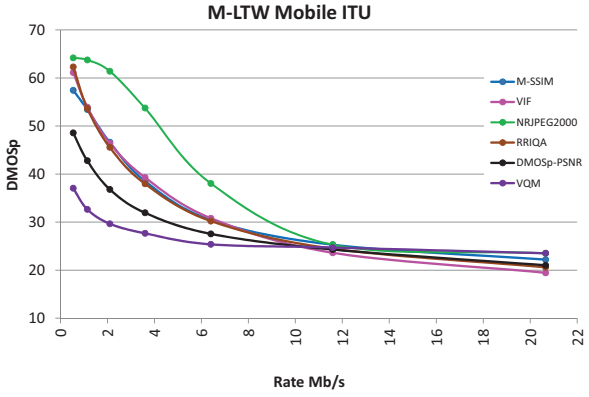


Figure 2.61: QAM comparison for Mobile ITU and M-LTW codec.

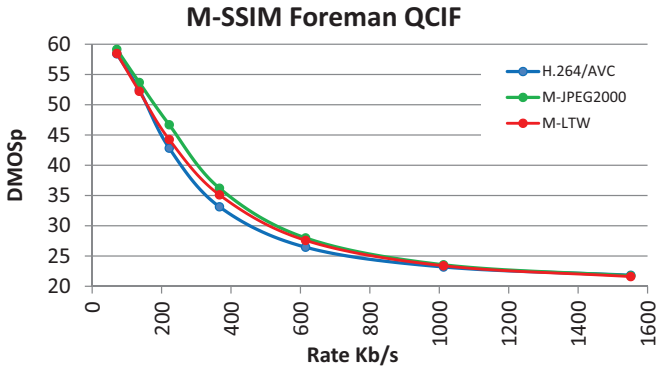


Figure 2.62: Encoders comparison for MSSIM - Foreman QCIF.

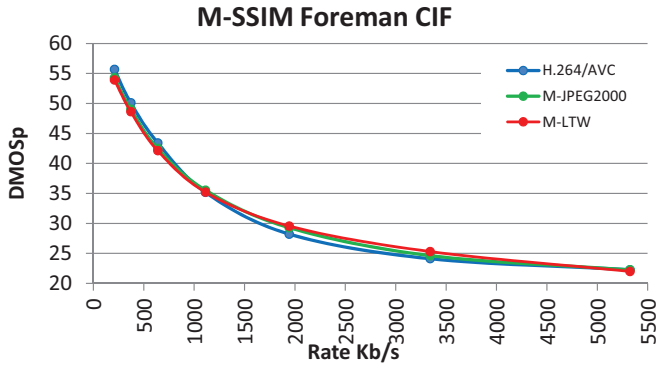


Figure 2.63: Encoders comparison for MSSIM - Foreman CIF.

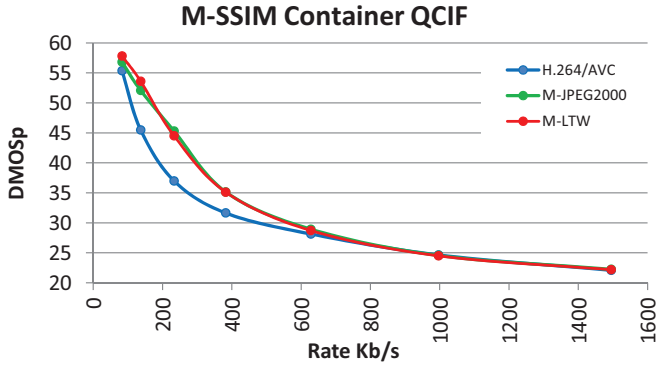


Figure 2.64: Encoders comparison for MSSIM - Container QCIF.

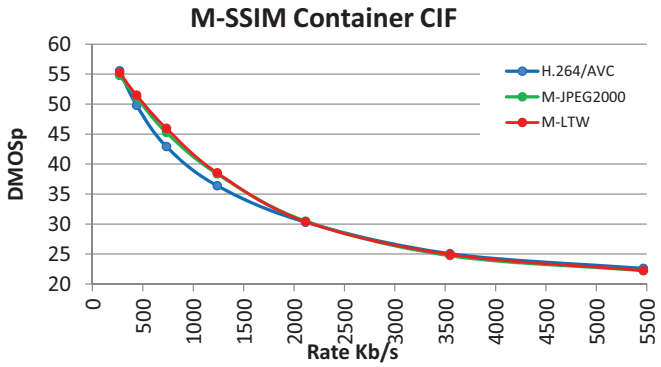


Figure 2.65: Encoders comparison for MSSIM - Container CIF.

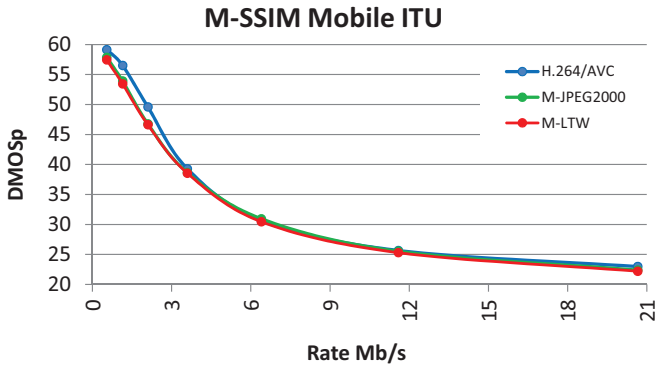


Figure 2.66: Encoders comparison for MSSIM - Mobile ITU.

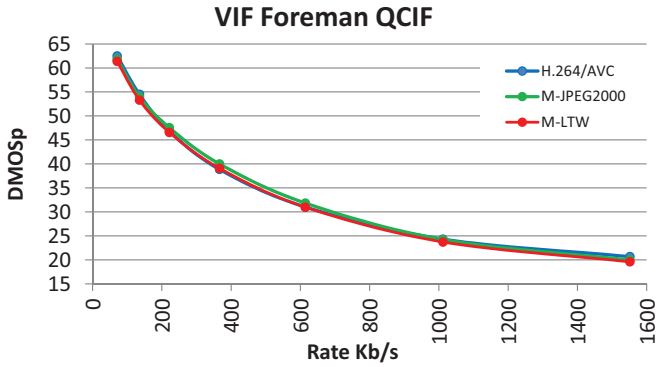


Figure 2.67: Encoders comparison for VIF - Foreman QCIF.

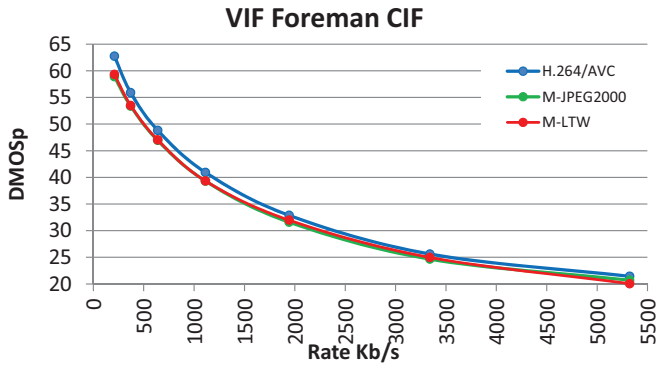


Figure 2.68: Encoders comparison for VIF - Foreman CIF.

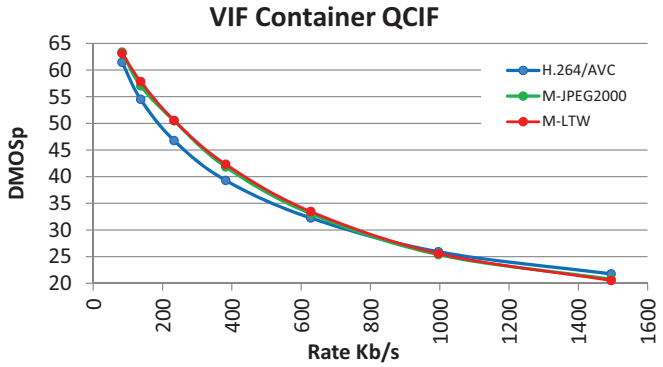


Figure 2.69: Encoders comparison for VIF - Container QCIF.

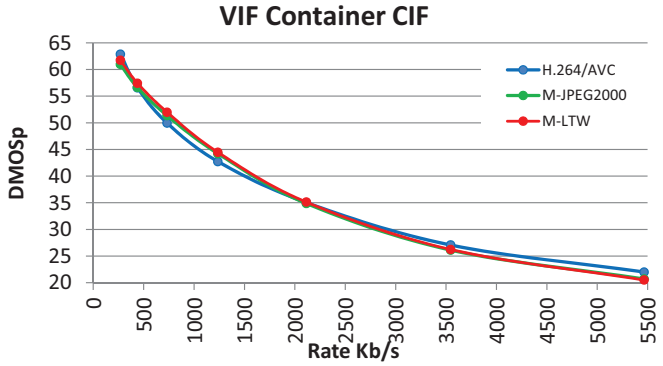


Figure 2.70: Encoders comparison for VIF - Container CIF.

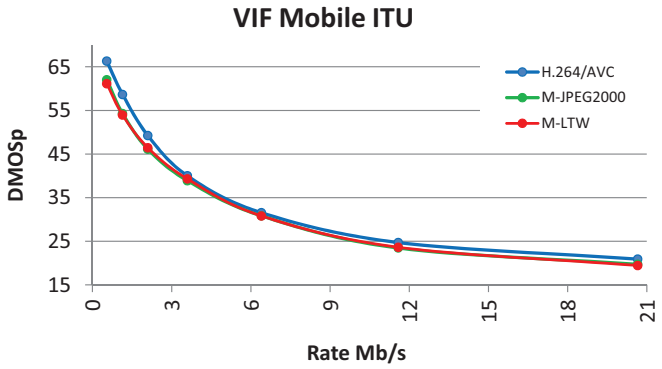


Figure 2.71: Encoders comparison for VIF - Mobile ITU.

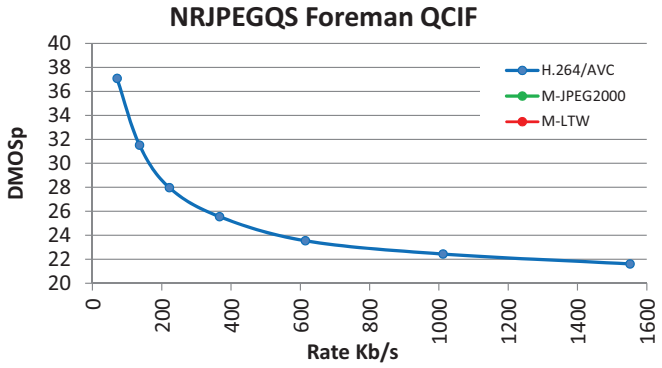


Figure 2.72: Encoders comparison for NRJPEGQS - Foreman QCIF.

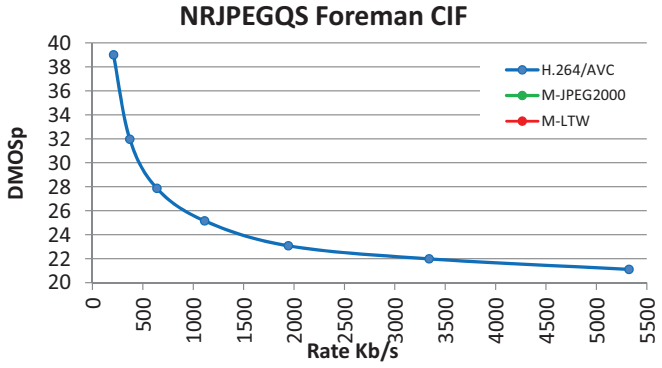


Figure 2.73: Encoders comparison for NRJPEGQS - Foreman CIF.

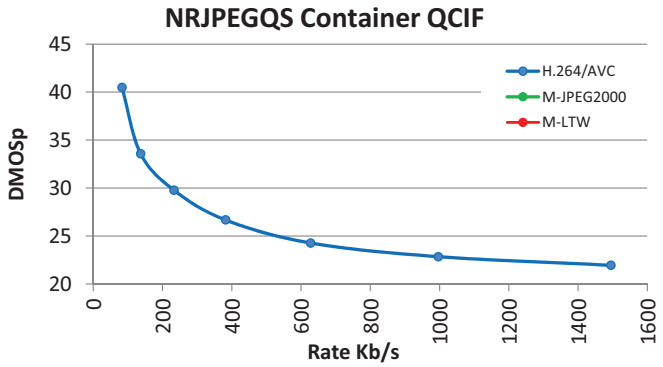


Figure 2.74: Encoders comparison for NRJPEGQS - Container QCIF.

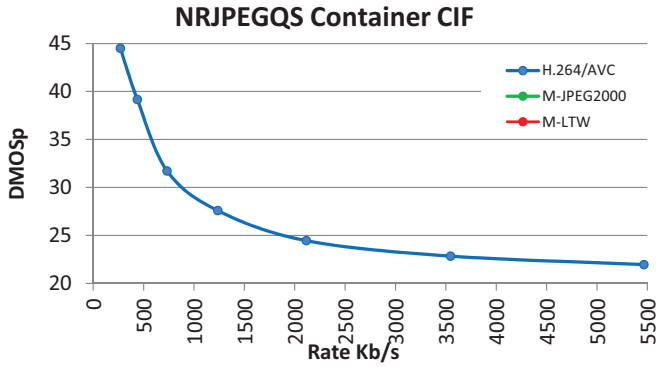


Figure 2.75: Encoders comparison for NRJPEGQS - Container CIF.

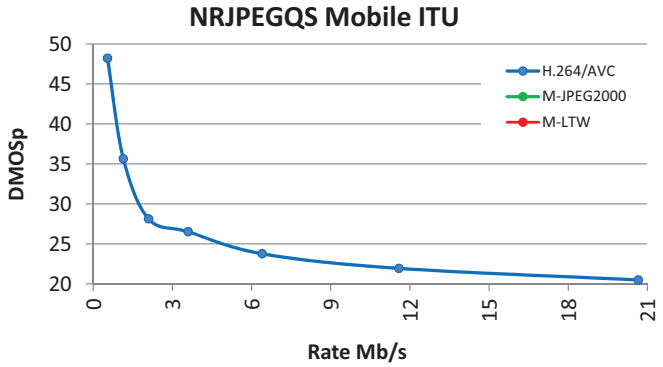


Figure 2.76: Encoders comparison for NRJPEGQS - Mobile ITU.

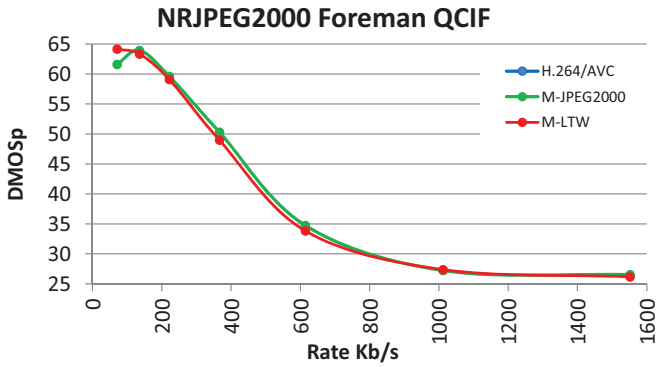


Figure 2.77: Encoders comparison for NRJPEG2000 - Foreman QCIF.

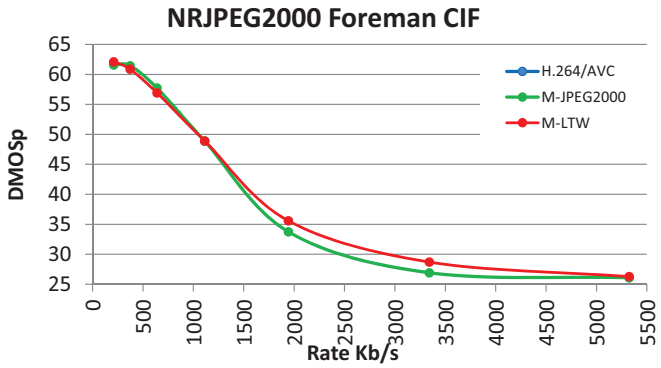


Figure 2.78: Encoders comparison for NRJPEG2000 - Foreman CIF.

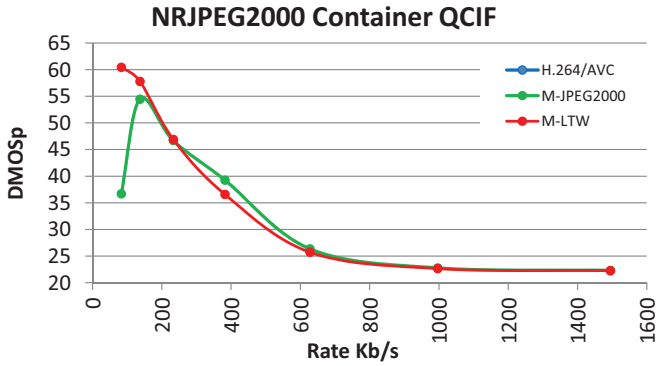


Figure 2.79: Encoders comparison for NRJPEG2000 - Container QCIF.

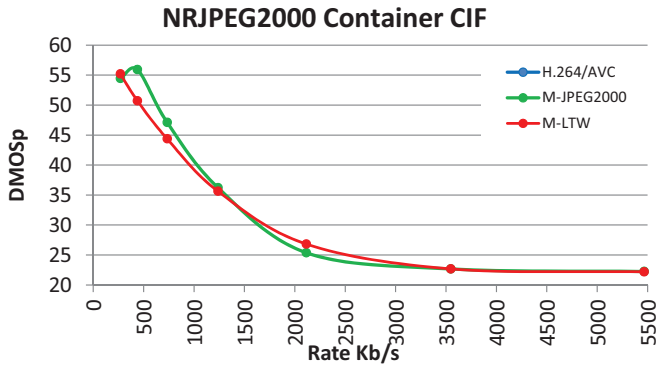


Figure 2.80: Encoders comparison for NRJPEG2000 - Container CIF.

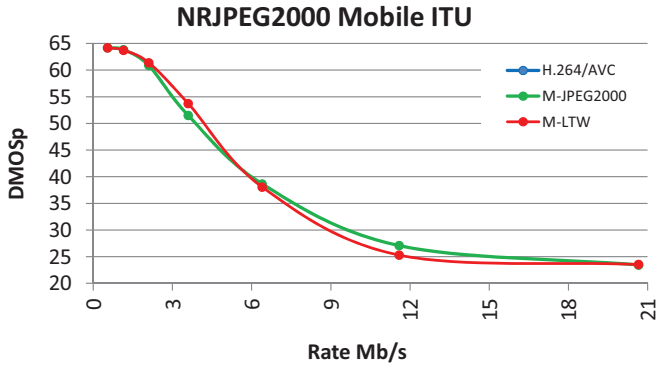


Figure 2.81: Encoders comparison for NRJPEG2000 - Mobile ITU.

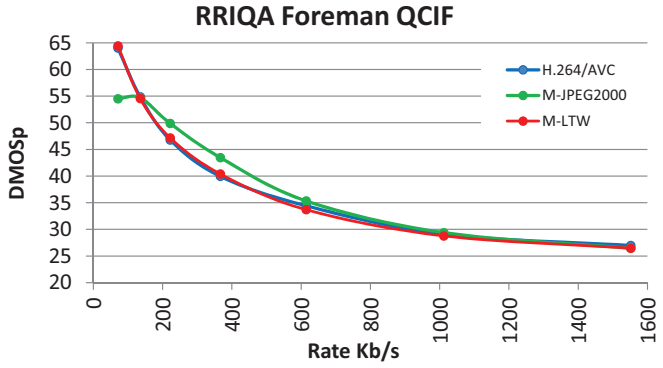


Figure 2.82: Encoders comparison for RRIQA - Foreman QCIF.

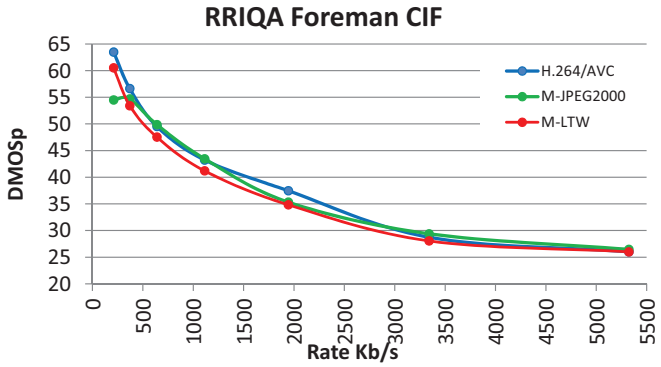


Figure 2.83: Encoders comparison for RRIQA - Foreman CIF.

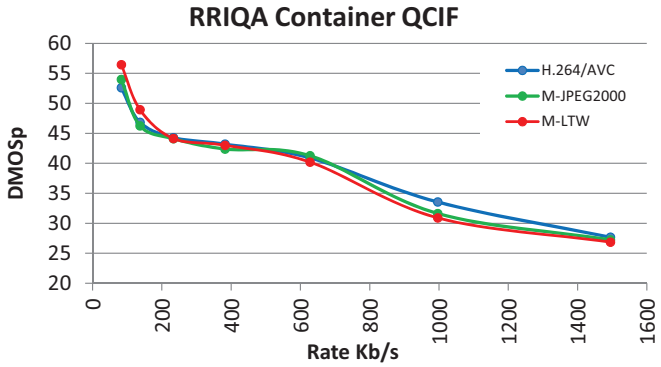


Figure 2.84: Encoders comparison for RRIQA - Container QCIF.

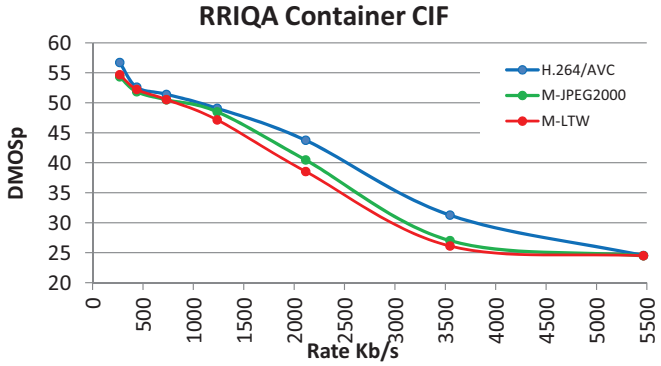


Figure 2.85: Encoders comparison for RRIQA - Container CIF.

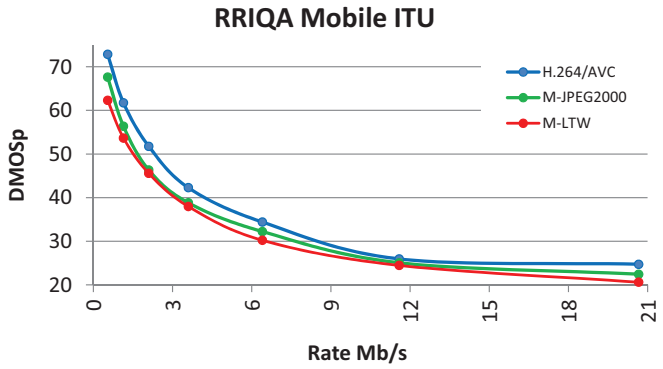


Figure 2.86: Encoders comparison for RRIQA - Mobile ITU.

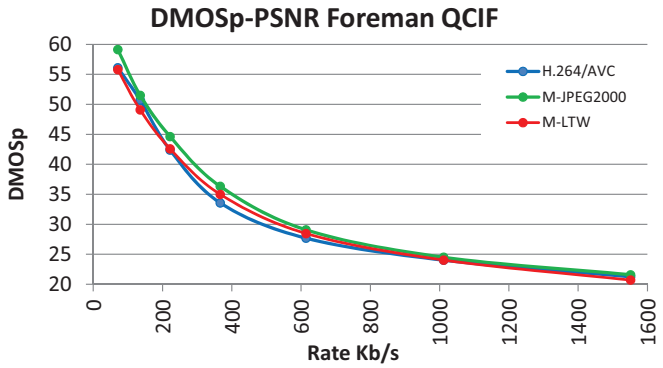


Figure 2.87: Encoders comparison for DMOSp-PSNR - Foreman QCIF.

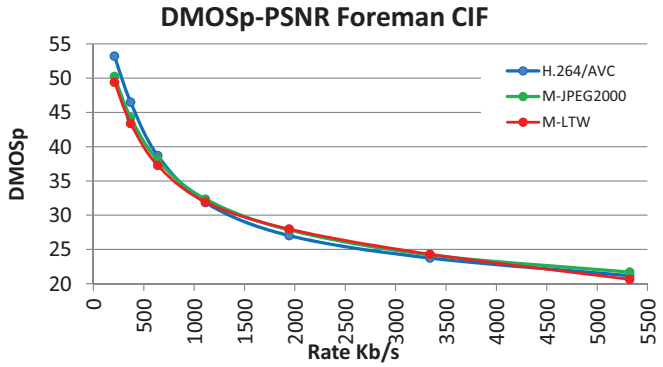


Figure 2.88: Encoders comparison for DMOSp-PSNR - Foreman CIF.

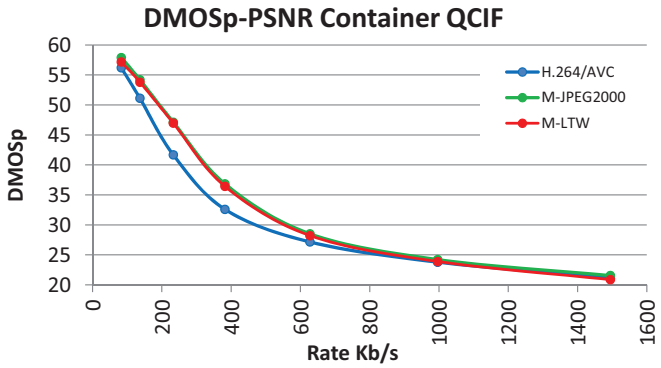


Figure 2.89: Encoders comparison for DMOSp-PSNR - Container QCIF.

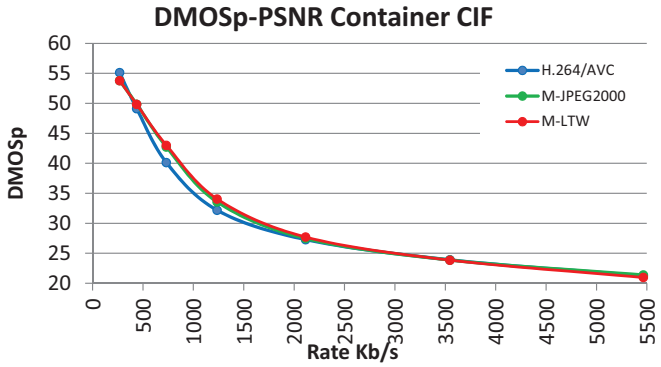


Figure 2.90: Encoders comparison for DMOSp-PSNR - Container CIF.

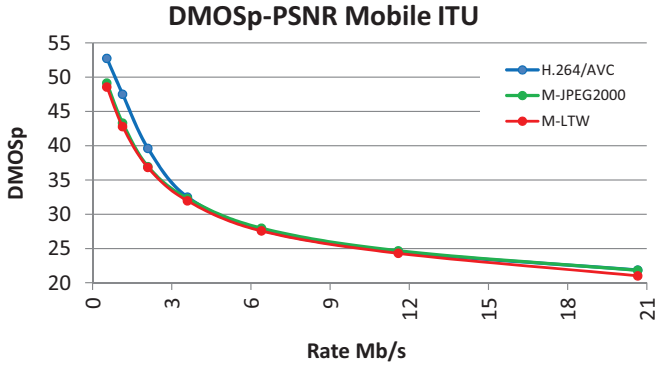


Figure 2.91: Encoders comparison for DMOSp-PSNR - Mobile ITU.

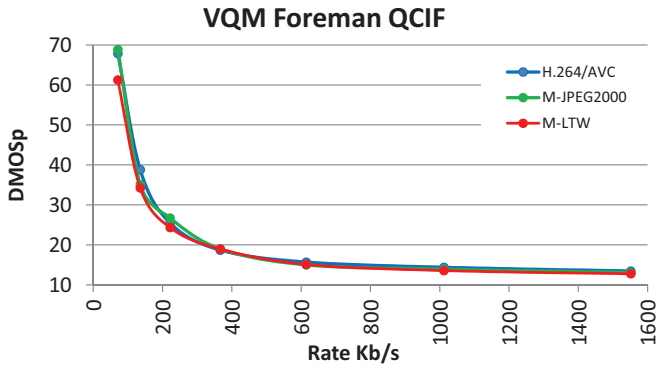


Figure 2.92: Encoders comparison for VQM - Foreman QCIF.

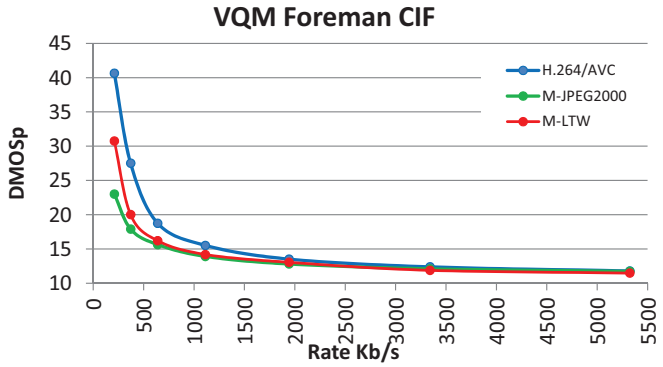


Figure 2.93: Encoders comparison for VQM - Foreman CIF.

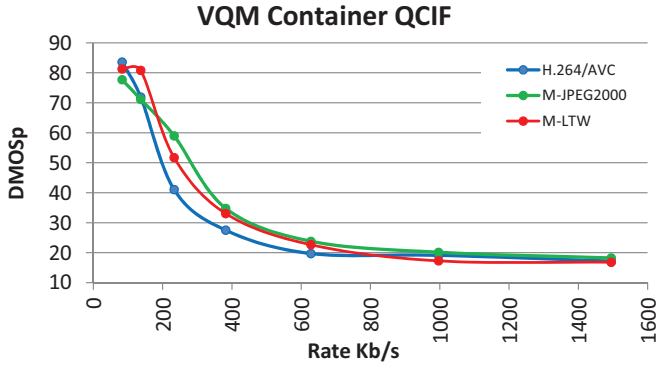


Figure 2.94: Encoders comparison for VQM - Container QCIF.

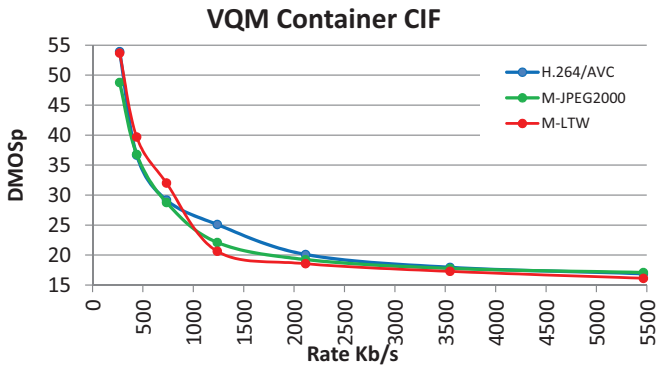


Figure 2.95: Encoders comparison for VQM - Container CIF.

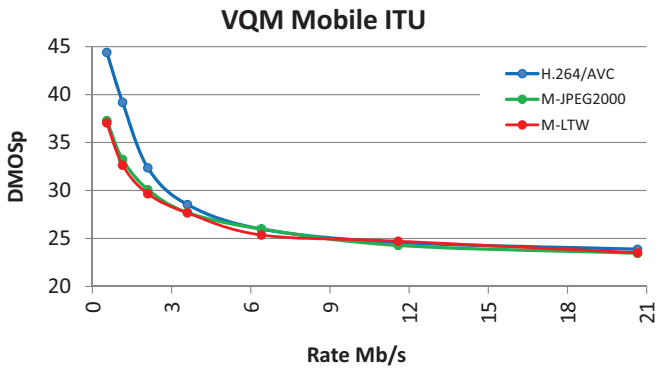


Figure 2.96: Encoders comparison for VQM - Mobile ITU.

Chapter 3

Perceptual Coding

Contents

Most of the HVS properties included in the design of perceptual based encoders are introduced in the quantization step. So we will briefly review the basics about quantization and then we will see, for DCT and DWT based encoders, how different strategies of perceptual quantization are included, mainly in the quantization step, but also in other encoder stages. As the most widely used characteristics of the HVS are contrast sensitivity and masking, we will take special care of them when reviewing the encoder proposals. We will also comment how encoders include other HVS characteristics in their designs, and how their performance results are presented or compared with other solutions.

3.1 Quantization

Quantization is the method or procedure followed to translate or reduce something from a continuous (infinite) set of values (such as real numbers) to another smaller discrete set of values (such as integers). The most basic and oldest form of quantization is rounding, where the infinite set of real numbers between two integers is assigned to either the lowest or highest integer. Gray and Neuhoff [230], comprehensively reviewed the most important quantization methods, so we will only review the basics to expose the quantization used in our proposals later. It is assumed that the sampling of the inifite set of values is uniform and the sampling rate is above the Nyquist rate so that there is no aliasing in the frequency domain.

Quantizers can be classified as memoryless or with memory. The former assumes that each sample is quantized independently with no prior knowledge of previous input samples whereas the latter takes them into account. Another classification is uniform or nonuniform quantizers.

Basically, to compress an image (a signal) means to quantize it; this means describing the image with less precision, and there are a lot of approximations to achieve that. After the set of pixels of an image has been transformed into coefficients due to the application of a frequency domain transform, the most basic form of quantization is to apply a uniform quantizer with a given quantization step ΔQ so that the quantized coefficient $\widehat{c}$ is represented by Equation 3.1, where c is the original coefficient and $\lfloor . \rfloor$ represents the rounding operator.

$$\widehat{c} = \Delta Q \left\lfloor \frac{c}{\Delta Q} \right\rfloor \tag{3.1}$$

Any quantizer can be decomposed into two distinct stages, referred to as

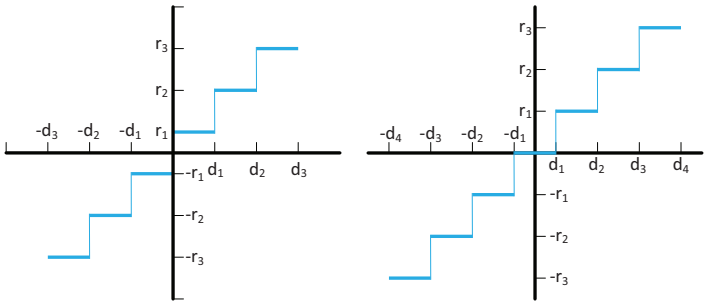


Figure 3.1: Staircase representation of uniform quantizers: a) midriser; b) midtreader

the classification stage (or forward quantization stage) and the reconstruction stage (or inverse quantization stage). In the example of the linear quantizer of Equation 3.1, this two stages are shown in Equation 3.2 for the forward step, and Equation 3.3 for the inverse step. The classification stage maps the input value to an integer quantization index u, and the reconstruction stage maps the index u to the reconstruction value $\widehat{c}$, which is the output approximation of the input value.

$$u = \left\lfloor \frac{c}{\Delta Q} \right\rfloor \tag{3.2}$$

$$\widehat{c} = u \cdot \Delta Q \tag{3.3}$$

Quantization is performed with a set of decision values d_j and a set of reconstruction values r_j such that if a coefficient c satisfies Equation 3.4, then the coefficient is quantized to a reconstruction value of r_j.

$$d_j \leq c < d_{j+1} \tag{3.4}$$

Figure 3.1 show the staircase representation of uniform midriser and midtreader quantizers. In a midtreader quantizer, the first step, usually centered on zero, has a reconstruction value of zero. In a midriser quantizer the first step has a nonzero reconstruction value and the decision interval for it begins at zero. Figure 3.2 shows the midriser and midtreader nonuniform quantizers staircase representation.

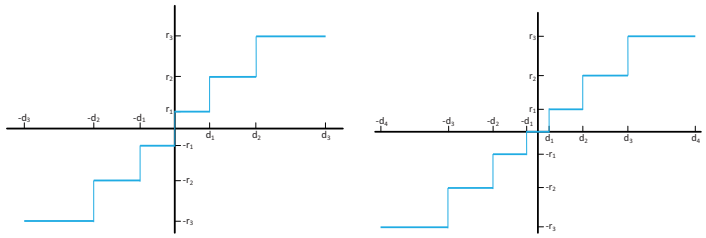


Figure 3.2: Staircase representation of nonuniform quantizers: a) midriser; b) midtreader

The uniform quantizer, also called scalar quantizer, is characterized for having all steps of equal size. So, to define a uniform quantizer we must provide the number of quantization levels, the step size, if it is midriser or a midtreader, and if it is symmetric or not. After frequency domain transforms, the resulting coefficients can be either positive or negative, so, our discussion is limited to symmetric quantizers, i.e., the input and output levels in the third quadrant are the negatives of the corresponding levels in the first quadrant (see the staircase representations).

The nonuniform quantizer has steps of different sizes depending on its design. So, to define a nonuniform quantizer we have to specify the input and output levels and these levels must be designed taking the probability density function of the input image into account.

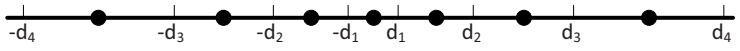


Figure 3.3: Line-segment representation of a nonuniform midtreader quantizer

Figure 3.3 shows the line-segment representation for the nonuniform quantizer. The reconstruction values are represented by a dot located in the center of each quantization step, but this is not mandatory; reconstruction values could be located at any point of the quantization step.

In spite of the type of the quantizer being used, a quantized output value (reconstruction value) is defined on a certain interval (inside decision levels) called quantization step, where any of the input values happens. The reconstruction value represents any of the input values inside the quantization step. Therefore, quantization is inherently a lossy process where the original input value may not be recovered.

Normally, in transform coding of natural images, a big distribution of

coefficients near zero are obtained. These near-zero coefficients corresponds to smooth areas in the image where less energy or variance is present. But each of these *low energy* coefficients should also be encoded, increasing therefore the size of the final bitstream. However, *low energy* coefficients increase the quality of the reconstructed image the least. So if we set one of them to zero, then it should not be encoded and hence it will not be recovered at all and the impact of this loss will hardly be noticed in the final quality of the reconstructed image.

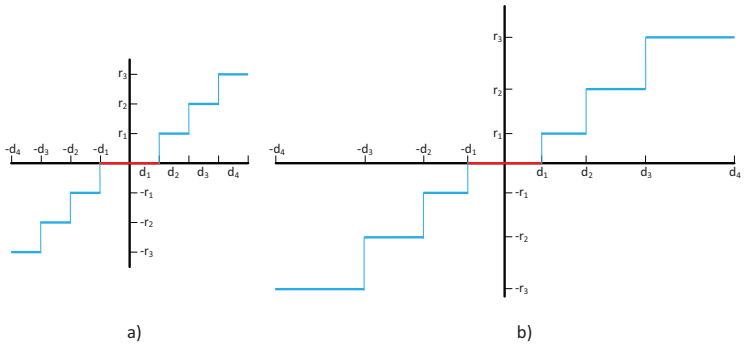


Figure 3.4: Deadzone quantizers: a) Uniform; b) Nonuniform

Taking into account the properties of transformed natural images, another type of quantizers, called deadzone quantizers, can be defined, see Figure 3.4. The deadzone is the region around zero where all coefficients will be set to zero so they need not be encoded and hence output value is zero. Except for the deadzone, the step size is constant, in uniform deadzone quantizers and variable for nonuniform deadzone quantizers. Such a nearly uniform quantizer has been specified in different image and video standards, as for example in JPEG2000.

The deadzone size could be variable as well, depending on the image or signal properties, so that an optimal deadzone size can be achieved for each image as a tradeoff between reducing the bit rate size and recovering the best quality.

The use of a nonuniform quantizer versus a scalar quantizer in encoders could improve the reconstructed quality of an encoded image. But the most difficult task is to design the optimal nonuniform quantizer. In [231, 232, 230], solutions and approximations for this optimal nonuniform quantization are presented from a R/D perspective. Other authors [233, 234] also discuss this topic and the use of other types of quantization strategies as vector quantization. A vector quantizer maps a set of input data (such as a block of image samples) to a single value (codeword) and, at the decoder, each codeword maps to an approximation to the original set of input data (a vector).

The set of vectors are stored at the encoder and decoder in a codebook.

Another way of performing a nonuniform quantization is companding. Due to the nonuniform characteristics of the input signal, some values are presented with higher probabilities than others. The companding technique, also known as logarithmic quantization, consists of the following three stages: compressing, uniform quantization, and expanding [230]. The compressing step applies a logarithmic compression characteristic to the input values so that the resulting probability density function is almost uniform. Then, a uniform quantization is applied. After quantization, the inverse transformation is applied to the quantized values, returning to the original non uniform probability distribution function.

The nonuniform quantization tries to minimize the MSE of the reconstructed signal by distributing the decision levels according to the statistics of the input random variable, in our case a natural image. When the statistics (mean, variance, etc.) of the input image differs from the ones that guided the construction of the nonuniform quantizer, then the performance of the quantizer is reduced. In other words, the same nonuniform quantizer will not have the same performance for all images or in all areas of the same image. This also occurs with the deadzone quantizers where a specific size of the deadzone performs better for a set of images while for another set, the best results are achieved with a different deadzone size.

3.2 Perceptual coding

In a non adaptive coding scheme, the coefficients are quantized using a fixed quantizer as exposed, and the quantized coefficients are usually entropy encoded to reduce redundancy.

Most of the encoders, regardless of the transform being used, determine a threshold value so that coefficients lower than it are set to zero. As mentioned previously, one cue is to get the best tradeoff between the loss in quality and the reduction of bit rate. As more coefficients are set to zero, the quality of the reconstructed image deteriorates. However, the way in which the image quality is affected depends not only on the number of non-zero coefficients retained but also on which coefficients are discarded, i.e., some coefficients are perceptually more important than others.

Another cue is to determine the appropriate quantization step size in each case. Once the coefficients have been thresholded, the remaining non-zero coefficients are quantized to reduce the number of bits. Over-quantization of coefficients corresponding to different spatial frequencies affects the

reconstructed image in different ways. For example, over-quantization of the low-frequency DCT coefficients causes blocking, while large quantization steps at higher frequencies lead to random noise becoming visible. Additionally, the phenomenon of spatial masking and luminance masking can also be taken into account in order to increase the quantization steps in specifics areas of the image. Since the HVS cannot detect quantization noises in highly textured as well as in smooth image regions [235, 236], appropriate elevation of the quantization step sizes will improve the coding efficiency of these textured regions without loss of perceptual quality.

For example, by analyzing the case of the JPEG standard (the analysis can be extrapolated to DWT encoders), we can see the need for additional subjective cues that guide the thresholding problem. In JPEG, images can be individually quantized via a Quantization Matrix (QM) that the standard does not fix. The main idea behind the QM is to be able to provide the best visual quality for a given image at a desired rate. The optimum QM for one image or a group of images could be not the optimum for others.

In the DCT quantization schema, each block coefficient c_{ijk} (i, j indexes the DCT frequency and k indexes the block) is quantized dividing it by ΔQ_{ij} and rounding to next integer, in Equation 3.5 the forward quantization stage is shown. Then the quantization error for each block in the DCT domain is shown in Equation 3.6, where the maximum possible error is $\Delta Q_{ij}/2$. In Equation 3.7, the reconstruction stage is shown. The reconstruction error E_{ijk} for each coefficient in block k is shown in Equation 3.8

$$u_{ijk} = \left\lfloor \left(\frac{c_{ijk}}{\Delta Q_{ij}} \right) \right\rfloor \tag{3.5}$$

$$\epsilon_{ijk} = c_{ijk} - u_{ijk} \cdot \Delta Q_{ij} \tag{3.6}$$

$$\widehat{c_{ijk}} = \Delta Q_{ij} \cdot u_{ijk} \tag{3.7}$$

$$E_{ijk} = c_{ijk} - \widehat{c_{ijk}} \tag{3.8}$$

The total quantization error obtained depends directly on the ΔQ_{ij} values for each frequency range. Errors will be different if we choose the same ΔQ_{ij} for all frequencies than if we choose a specific ΔQ_{ij} for each one. So, the idea may be to find the appropriate quantization value (ΔQ, Quantization step (Qstep)) for each frequency range, represented by each DCT coefficient that minimizes the total quantization error.

Many approaches can be found in the literature to build the optimum quantization matrix. One of the first was [237] where authors performed a set of psychophysical tests to set for each frequency ij the threshold T_{ij}, i.e., the smallest coefficient that produces a visual signal. Having such a threshold, to ensure that each error is below threshold, i.e., invisible, the maximun quantization error $\epsilon_{ijk-max}$ should be this threshold. From equations 3.5 and 3.6, due to the rounding operation, the maximum possible error is Equation 3.9 and then the quantization step for each frequency range can be set as in Equation 3.10

$$\epsilon_{ijk-max} = \frac{\Delta Q_{ij}}{2} \tag{3.9}$$

$$\Delta Q_{ij} = 2T_{ij} \tag{3.10}$$

Building a QM with Equation 3.10 assures, trusting in the subjectively derived thresholds, that quantization errors will be unnoticed. But the thresholds must be calculated again for each image because as we know, image content can vary the perception of errors. So, at the end, obtaining the optimum QM, which is image dependent, is a time consuming process where subjective tests must be followed in order to obtain the optimum frequency thresholds for a particular image. If we use averaged thresholds over a set of images, then the QM is sub-optimal for these images.

Furthermore, this QM building procedure unfortunately does not take into account several important perceptual issues such as:

- Luminance masking; where variations in the DCT thresholds should be made to account for local mean luminance.
- Contrast masking; that modifies the threshold for particular DCT coefficients, those with same frequency and orientation as the masker.
- Equation 3.10, assures only that each individual error is below threshold, but does not assure that all possible errors jointly are under threshold.
- If all errors are under threshold and contrast masking is not included, a specific bit rate is obtained, but it is possible that, taking into account contrast masking, bit rate could be further reduced maintaining the perceptual quality. The problem is then to find a perceptually rate-control algorithm.

The quantization processes, followed with an entropy encoding stage, can reduce most of the statistical redundancy from an image. Adaptive quantization tries to adjust the quantization strategy to the input image statistics based on the coefficients Probability Density Function (PDF), adapting the quantization steps for the input image or parts of it. Reviews and explanations of several adaptive approaches can be found in [238, 239]. These adaptive quantization approaches, although based on statistics of the images being quantized, do not include steps, stages or algorithms, based on the knowledge of how the HVS processes images, that could solve the four mentioned perceptual issues.

A number of methods have already been proposed that include certain psychovisual properties of the HVS (frequency sensitivity, luminance dependence, and masking effects), into image coding and compression schemes that try to solve some or all of those issues.

In the next sections we will briefly review the schemes or strategies used to include the CSF and the Masking properties of the HVS in image and video encoders. Then we will review some of the most relevant works that use these strategies, focusing in 1) how the HVS properties have been included in the encoders, 2) how the adaptive quantization is performed, 3) the frequency transform being used (as most of the works are specifically designed for one type of frequency transform, i.e., DCT, DWT, or other transforms) and 4) how the different proposals present and compare their performance comparisons.

3.2.1 Contrast and CSF

The Contrast Sensitivity Function (CSF) measures the response of the HVS to different frequencies, i.e., quantifies how well the HVS perceives a contrast at a given spatial frequency. Another perspective of CSF is that it is the reciprocal of the contrast necessary for a given frequency to be perceived. In this section some of the most important CSF models are cited, an overview of how the CSF has been used is done, and we expose which is the CSF model we use jointly with the parameters defined in our proposals.

3.2.1.1 CSF models

CSF has been widely used in the literature to include the HVS sensitivity to contrast into many encoder and QAM proposals. Different variations or models for the CSF can be found. Using the assumption that HVS is isotropic, most authors modeled the HVS with a Modulation Transfer Function (MTF), which is given by Equation 3.11 where f is the radial frequency in cycles/degree of the subtended visual angle, and a, b, c, and d are constants.

$$H(f) = a(b + c \cdot f)e^{-(c \cdot f)^d} \tag{3.11}$$

One of the first CSF proposals was made by Mannos and Sakrison [240], after conducting a series of psychophysical experiments on human subjects. In spite of being one of the first proposals, it is the most cited and adapted one, and most researchers used this model jointly with the DWT transform, so it has been adopted in Part II of the JPEG2000 standard, [241, 242, 243, 244] and is the one used for our proposals.

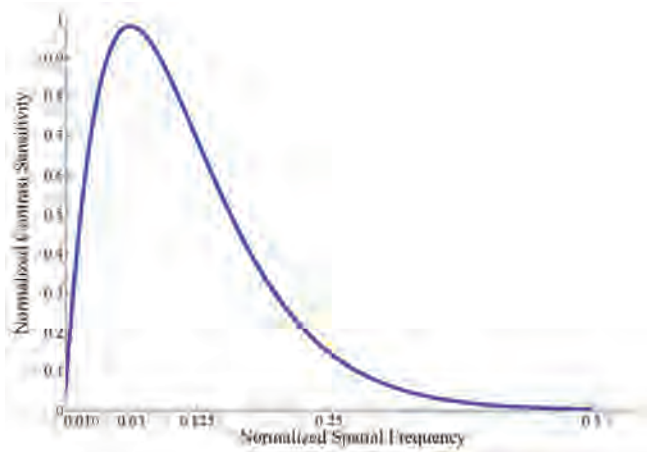


Figure 3.5: Contrast Sensitivity Function

$$H(f) = 2.6(0.0192 + 0.114f)e^{-(0.114f)^{1.1}} \tag{3.12}$$

Equation 3.12 shows the Mannos and Sakrison model, where spatial frequency is usually measured in cycles per optical degree (cpd). This model has a peak at approximately 8 cpd.

In Figure 3.5, the CSF curve obtained with Equation 3.12 is depicted. It characterizes luminance sensitivity as a function of normalized spatial frequency. The y axis corresponds to the contrast sensitivity ($CSF = 1/Contrast\ threshold$), and the x axis corresponds to the normalized spatial frequency that represents half of the spatial sampling frequency, due to the Nyquist theorem.

As shown, CSF is a bandpass filter, which is most sensitive to normalized spatial frequencies between 0.025 and 0.125 and less sensitive to very low and very high frequencies. The reason why we can not distinguish patterns with high frequencies is the limited number of photoreceptors in our eye. CSF

curves exist for chrominance as well. However, unlike luminance stimuli, human sensitivity to chrominance stimuli is relatively uniform across spatial frequency.

$$H(f) = (0.2 + 0.45f)e^{-0.18f} \tag{3.13}$$

$$H(f) = (0.31 + 0.69f)e^{-0.29f} \tag{3.14}$$

$$H(f) = 0.246(0.1 + 0.25f)e^{-0.25f} \tag{3.15}$$

Other CSF models have been proposed since, such as Nill's model in [245] that is used with the DCT transform. Nill's model corresponds to Equation 3.13, which has a peak at 5 cpd. In [246], Ngan et al. propose the model of Equation 3.14 with a peak at 3 cpd. These two models are adaptations of the Mannos and Sakrison model, obtained by multiplying it by a $A(f)$ function that shifts the peak and adapts the model to be used with the DCT; for more details see [245, 246, 247]. The same approach is used in [247] where Chitprasert et al. propose a CSF weighting matrix for the DCT coefficients. Their model corresponds to Equation 3.15, which has a peak at 3.75 Cycles per Degree (cpd). Also Chandler and Hemami, in [248], propose a model for obtaining the contrast thresholds to be used as base sensitivity thresholds in DWT encoders.

Two different ways to measure a CSF [249], either by threshold detection or by intensity/color matching experiments.

In the first case, the contrast of a Gabor patch displayed on top of a uniform background is reduced until it can no longer be distinguished from the background. At that point, the perception contrast threshold is reached $C_T(f)$. This threshold is commonly referred to either as base sensitivity threshold, or base threshold. Compression methods that only take into account these base thresholds are referred as at-threshold or sub-threshold methods. The inverse of the contrast at this threshold is defined as the sensitivity for that frequency $S(f)$. The resulting curve is normalized to a maximum of 1.0 for compression applications as only the relative sensitivity is important.

The second method displays a striped pattern of a specific frequency and side by side a patch of uniform intensity on top of a uniform background. The observers have to adjust the intensity of the uniform patch until it matches it perceived intensity of the striped patch. In this case, the contrast sensitivity is directly proportional to the adjusted intensity. In this second experiment, the test patterns are always clearly distinguishable from the background. That

means the experiment operates at a supra-threshold level and must be designed carefully [249].

The sub-threshold and supra-thresholds experiments to obtain a CSF model result in different CSF shapes due to the nonlinear characteristics of the HVS. This is a particularly important point in the context of image compression. Experiments at detection threshold are used for the CSFs because they are more stable and easier to measure, but they are valid only for sub-threshold compression. However, it is certainly valid to measure artifact visibility at near-visually-lossless rates [249].

3.2.1.2 Including the CSF

The base sensitivity thresholds, obtained from the different models, are used to fully quantize, i.e., to remove the transformed coefficients that are below threshold. These coefficients are supposed to correspond to perceptually redundant information and can be discarded. This reduces the bit rate needed to encode the image without loss of perceptual quality. Other authors, instead of obtaining the thresholds directly from a previous CSF model, perform a series of subjective tests to detect contrast Just Noticeable Differences (JND), and based on their findings, they provide a model to obtain those thresholds or they provide a sub-threshold weighting matrix.

These thresholds are arranged in perceptual quantization matrices, also called CSF weights or perceptual weighting matrices, where each value in the matrix corresponds to one frequency interval. Depending on the transform being used, the value is applied either to a DCT block or to a DWT subband. So, in perceptual coding and compressions schemes, these CSF weights can be exploited mainly two ways.

- In the first method, the CSF weights are used to modify the transformed coefficients before and after quantization.
- In the second method, the CSF weights are used to modify only the distortion function of the rate-distortion control algorithm. This is a decoder independent approach.

The first method is shown in Figure 3.6. In a DWT encoder, the CSF weights are introduced in the encoder using the Invariant Scaling Factor (ISF) weighting strategy explained also in [249]. Once the corresponding weights for each frequency subband are obtained, they are introduced after the wavelet filtering stage and before the quantization stage. The weighting consists on the

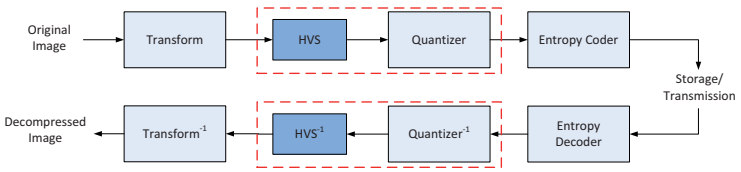


Figure 3.6: CSF weights included in the encoding and decoding chain.

multiplication of the wavelet coefficients in each frequency subband by the corresponding weight. At the decoder, the inverse of this weight is applied. The CSF weights do not need to be explicitly transmitted to the decoder. This stage is independent to the other encoder modules (wavelet filtering, quantization, etc). Using this approach the HVS and Quantizer blocks of Figure 3.6 are jointly obtained by Equation 3.16 if the quantizer is a scalar one, being w_{csf} the weight that corresponds to the coefficient c.

$$\widehat{c} = \Delta Q \left\lfloor \frac{c \cdot w_{csf}}{\Delta Q} \right\rfloor \tag{3.16}$$

The second method is used in codecs like the JPEG2000 standard Part II, where the CSF weights are introduced as a Visual Progressive Single Factor (VPSF) weighting, replacing the MSE by the CSF-WMSE and optimizing system parameters to minimize WMSE for a given bit rate. This is done in the post-compression rate-distortion optimization algorithm where the WMSE replaces the MSE as the cost function which drives the formation of quality layers.

Both methods are referred in [249] as non adaptive CSF implementations, because for each DWT subband or DCT block the same invariant scaling factor is applied to each coefficient in the subband/block. There is no adaptivity for the different spatial frequencies that are present in the different spatial locations of the image. In DWT approaches this can be easily done because it captures not only frequency information but also location information. In DCT based encoders the spatial adaptivity is performed for each DCT block that correspond to a specific location in the image.

Regardless of the chosen approach, discarding perceptual redundant information is the main idea behind sub-threshold coding, and it is used in most of the HVS inspired proposals, as we will see later in section 3.2.3.

3.2.1.3 Distance and resolution

As exposed, the frequency in equations 3.12 to 3.14 is usually expressed in cycles per optical degree. For a visual angle of one optical degree, the size of the viewed scene covered by this optical degree, depend on the distance to the viewed scene. If the size increase then more cycles per degree fit in. So, we have to talk about spatial frequency not in the world but on the back of your eye, on the retina. Suppose a sine wave grating such as of Figure 2.15, the thickness of bars in a grating is called spatial frequency - how frequently bars occur across space. If lots of bars occur across a particular distance, then the grating has very thin bars and is said to have high spatial frequency, like on the right side of Figure 2.15. If very few bars occur across the same distance, then the grating has thick bars and is said to have low spatial frequency, like on the left side. The black/white colors appear less intense in the right pattern than those on the left due to the reduced sensitivity of the HVS for high-frequencies The amount of bars that one can perceive in such an image depends on the distance.

Figure 3.7: Distance and visual angle

The size of an object on the retina is measured by the size of the angle it subtends (called visual angle). Figure 3.7 shows shows that an object 1 cm tall at a distance of 57 cm subtends a visual angle of 1°.

$$f\left(\frac{cycles}{degree}\right) = f_n\left(\frac{cycles}{pixel}\right) \times f_s\left(\frac{pixels}{degree}\right) \tag{3.17}$$

Equation 3.11 is usually expressed in cycles/degree, and Equation 3.17 establishes the relationship between pixels and cycles. f_n is the normalized spatial frequency in the range from 0 to 0.5, the sampling frequency f_s is the number of pixels within 1° degree, which depends on the distance as the number of pixels that fits in 1° degree of visual angle increases with distance. So, the sampling frequency in pixels/degree, f_s, is usually obtained via Equation 3.18 where v is the viewing distance in meters and r is the resolution in dots or pixels per inch.

$$f_s = \frac{v \cdot \tan(0.5°) \cdot r}{0.0254} \tag{3.18}$$

If the image is critically downsampled at the Nyquist rate, 0.5 cycles/pixel are obtained. This means that the maximum frequency represented in the signal, measured in cycles per degree, is f_{max} (see Equation 3.19).

$$f_{max} = \frac{fs}{2} \tag{3.19}$$

Therefore, although the unit cycles/degree is independent of the visual distance, from the previous definitions we see that in order to obtain the contrast thresholds for a specific CSF model, we must take into account the visual distance, i.e., the thresholds for a specific frequency depend on the distance. This is handled with two different approaches in the literature.

The first one, the most accurate one, is to provide several weighting matrices, one for each specific visual distance, or to provide a parametrized weighting matrix that depends on the visual distance.

The second one, the most restrictive, and the one that we will use in our proposals, is to assume the *worst viewing conditions* [249].

As known, the ability to detect some distortions in encoded images decreases with the distance, so what we mean by *worst viewing conditions* are those that use a high resolution display (or printed resolution) viewed as close as possible, so that it is possible to detect more distortions. The rationale behind this is that as we go far from the image, some of the distortions introduced by the perceptual quantization matrix may disappear, and when we approach the image they become visible again. Besides, when a viewer is told to inspect an image in order to detect distortions in it, he subjectively approaches the image as much as his visual accommodation allows him. How much he approaches varies also with the display dimensions.

So, to calculate the sub-threshold quantization matrix under the *worst viewing conditions*, two parameters must be fixed: the display resolution r in pixels per inch and the visual distance v in meters. Using Equation 3.18, we will obtain the sampling frequency for those conditions, and hence, with Equation 3.19, we obtain the maximum frequency f_{max} in cycles/degree that is used as upper bound in Equation 3.11.

The distance is supposed to be the minimum distance to perform visual accommodation, but for most of the encoding proposals, a visual distance of 3 to 4 times the image height is used, but there is no consensus, and the visual distance parameter used by each author is slightly different. The American National Standards Institute (ANSI) Standard for Visual Display Terminal Workstations (ANSI/HFS 100-1988) presented their recommendations where the minimum desirable distance is 12 inches (30.5 cm.). Although it is

possible to approach farther, up to the physiological limits, to an image, users normally use this distance. In fact, some authors, like Watson et al. [106] use this limit. We will therefore also use 12 inches as visual distance in our proposals.

The other parameter is the display resolution. As told in previous chapters, the maximum spatial frequency that is able to detect the HVS is about 60 cpd. Some studies increase this limit to 65 cpd. This spatial frequency can be obtained two ways, as derived from Equation 3.18, by increasing the distance or by increasing the display resolution. If we fix the distance, for example to 12.23 inches, then a display that is able to reproduce 64 cycles/degree should have a display resolution of 600 dpi (or ppi) . This is a common resolution for printed material, which can be even higher, but nowadays maximum display resolutions are around 445 to 538 ppi in some high segment mobile phones with retina display, up to 288 ppi in digital cameras, up to 265 ppi in E-ink screens, up to 110 ppi for High Definition (HD) TV and finally up to 204 ppi in 16:10 wide aspect for a 3840x2400 resolution desktop display. In the time of writing this text, it is a big competition of develop the display with the highest pixel density, a display manufacturer announced a 7-inch tablet with 600 ppi.

So for our proposals we set the parameters in 300 ppi and 12 inches what produce 32.01 cycles/degree of maximum frequency for Equation 3.12. This moves the peak of the CSF curve to 4 cpd.

3.2.2 Masking

Visual masking is a perceptual phenomenon, see section 2.2, where artifacts are locally masked by the image acting as a background signal. Two general types of visual masking are mostly used in perceptual coding and the common terms to refer to it are luminance masking and contrast masking. There are, in the literature, several variations of the basic procedures revised here, as we will see in Section 3.2.3. So, in this section we will briefly overview the main ideas and basic procedures of how these masking effects could be included into the encoders.

Both types of masking work in an adaptive way, moving along the image (typically block wise) and applying the masking effect for each location with different granularity, depending on the complexity of each proposal. In DCT encoders, this is applied for each DCT block or even for each DCT coefficient, while in DWT encoders some authors apply it only for each DWT decomposition level or subband, while others perform an additional block segmentation in order to apply it on a finer scale.

The first one, luminance masking, also called brightness adjustment or light adaptation, is quite easy to implement and is mainly included in the thresholding stage, modifying the base sensitivity thresholds depending on the luminance of the scene at each spatial location.

For the second one, the common term of contrast masking can be further divided as in [111] by two types of masking effects. Contrast masking or spatial masking and texture masking or energy masking.

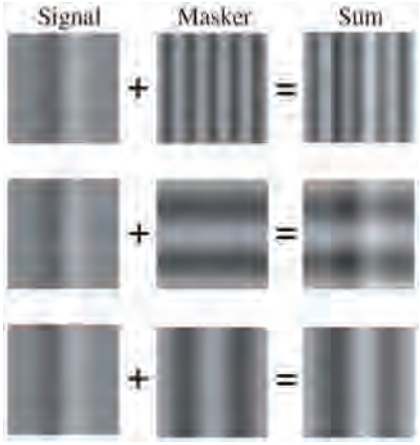


Figure 3.8: Contrast masking. The signal is masked depending on the orientation and frequency of the masker.

Contrast masking or spatial masking is used when referring to the spatial frequency and orientation differences between the masker (the image) and the signal (the artifact). When both signals have the same orientation, the masker hides the signal quite well but when orientations are different then the signal is clearly shown. See image 3.8 from [111].

Figure 3.9: Texture masking example.

Texture masking or energy masking refers to the fact that a concentrated distortion signal is easily recognized in a smooth and homogeneous zone, while it is somehow hidden in an active region. In image 3.9, the distortion is clearly

visible in the smooth area while it is more difficult to detect in the textured one (lower-left) [111].

The basic idea behind contrast masking and texture masking is to modify the base sensitivity thresholds at the location to be applied, depending on the frequency of the signal and masker for contrast masking and depending on the amount of energy (texture, entropy, etc...) for the texture masking. In texture masking, the main differences between proposals is in the way that energy or texture is calculated. Some authors perform a block or region segmentation based on these values, for example in texture, smooth, or edge block/regions. Others apply it even at a finer scale for each coefficient, taking into account its neighborhood.

3.2.2.1 Luminance masking

The terms *luminance* and *brightness* are commonly used interchangeably, but straightly speaking, the luminance value is a physical measure, while the term *brightness* is a subjective descriptor that cannot be measured. Besides, we find the term *grayscale*, which refers to the luminance component of a digital image. For instance, an 8-bit grayscale value of zero means total darkness (black color), or the lowest luminance, while the maximum 8-bit grayscale value of 255 means bright white, or the highest luminance. The concept of luminance is illustrated in Figure 3.10, where the magnitude ΔL is the one at which the perturbation is just visible.

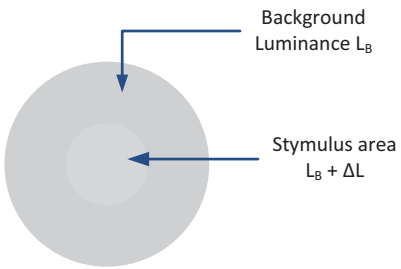


Figure 3.10: Background luminance with a ΔL visibility threshold

Experiments show that the visibility threshold ΔL is a function of the background luminance L_B and it increases almost linearly with L_B. This is known as Weber's law (Equation 3.20), which indicates that human eyes are less sensitive to errors in the bright areas because ΔL must have a higher value in order to maintain the Weber fraction constant. By contrast, in dark areas, where L_B is small, a small amount of ΔL is sufficient to maintain the constant, i.e., in dark areas, a small increment of luminance is perceptible while in

bright areas, the luminance increase must be higher in order to be noticed.

$$\frac{\Delta L}{L_B} = constant\ (Weber\ \ fraction) \tag{3.20}$$

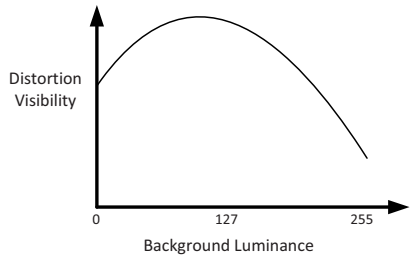


Figure 3.11: Distortion visibility vs. background luminance.

Weber's law is generally accurate over the normal range of middle-low to high luminance values. However, in very dark areas, it has been reported that the Weber fraction tends to increase with decreasing background luminance values, i.e., the human eye's sensitivity to distortion also decreases in a very dark area, see Figure 3.11.

Detection threshold for a luminance pattern typically depends upon the mean luminance of the local image region: the brighter the background, the higher the luminance threshold [164]. Ahumada and Peterson and later Watson too, [250, 251] proposed the formulas for the threshold T_{ij} values as a function of the mean luminance for a DCT block where i and j index the block, taking into account also the main luminance of the display. These formulas can be, however, approximated by a power function as in [164].

$$t_{ijk} = t_{ij} \left(C_{00k} / \bar{C}_{00} \right)^{a_T} \tag{3.21}$$

This approximation corresponds to Equation 3.21 where the threshold for each ij coefficient of block k is represented by t_{ijk} and the exponent of the power function is chosen with the same value (0.649), as in [251]. Note that the luminance masking can be suppressed by setting $a_T = 0$, which consequently controls the degree of masking. The value t_{ij} is the DC frequency sensitivity of coefficient ij, C_{00k} is the DC coefficient for block k, and $\bar{C}_{00}$ is the average among the DC coefficients in a picture.

$$a_l\left(\lambda, \theta, i, i\right) = \left(\frac{v_{\lambda max, LL, i', j'}}{v_{mean}} \right) \tag{3.22}$$

The luminance thresholds also have been applied in DWT encoders, like in [252] where Equation 3.22 is used to calculate the luminance masking adjustment, where v_{mean} is the mean luminance constant corresponding to the LL subband (128 in a 8-bit unsigned image). The $v_{\lambda max,LL,i',j'}$ value is the value of the DWT coefficient in the LL subband, which spatially corresponds to the coefficient i, j in the (λ, θ) subband, being λ the wavelet decomposition level and θ the subband orientation. The correspondence between spatial coefficients is obtained by equations 3.23 and 3.24 where $\lfloor . \rfloor$ represents the rounding operator and λ_{max} is the number of decomposition levels.

$$i' = \left\lfloor \frac{i}{2^{\lambda_{max}-\lambda}} \right\rfloor \tag{3.23}$$

$$j' = \left\lfloor \frac{j}{2^{\lambda_{max}-\lambda}} \right\rfloor \tag{3.24}$$

A mean luminance of 128, half the luminance range for 8-bit representation, is commonly used for obtaining the luminance thresholds. Other authors perform subjective tests in order to establish the luminance thresholds for each frequency subband. The problem of using a unique mean value to obtain the luminance thresholds, as stated in [253], is that these techniques are not adaptive enough. Suppose a mean grayscale value of the image of 90, then most of the blocks the image will have DC values below 128, resulting in under-utilization of the luminance masking. Over-quantization may be applied also if the mean luminance is 160, for example, instead 128. Therefore, some authors like [253] use different factors depending on the region in the scale 0 to 255 where the luminance threshold is calculated. The same study also provides results from inspecting natural images that provide a large range of mean image luminance values, varying from 78 to 164.

3.2.2.2 Contrast and texture masking

Contrast masking refers to the reduction in the visibility of one image component by the presence of another. In compression applications, the image itself acts as background, reducing the visibility of quantization noise in those areas of the image with same frequency and orientation or with high texture or energy content. In smooth areas of the image, for example in a clear sky at high compression levels, compression artifacts are more visible than in other textured areas.

The design of a compression system that exploits visual masking effects

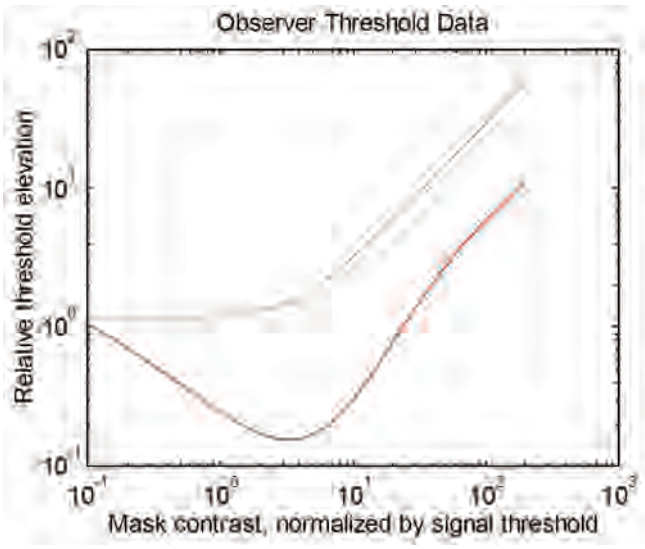


Figure 3.12: Psychophysical data for the threshold vs. masking contrast.

is based on psychophysical data for the threshold vs. masking contrast [254]. Those experiments provide the typical light adaptation curve shown in Figure 3.12 [255] where two functions represent two types of masking patterns. The upper curve corresponds to white noise with a narrow band and uncorrelated phase, while the bottom function corresponds to a sine wave entirely correlated in phase.

For the noise mask, the threshold initially stays constant but then the slope increases until it reaches a constant slope near 1.0 in the log-log plot for high noise contrast. For sine masking, there is an additional region, called the dipper effect region, where the threshold is reduced. In this region facilitation occurs. In this type of masking the slope of the rest of the curve is slightly lower, typically around 0.7. Natural images actually contain maskers between these types of masks.

As a consequence of the masking produced by the image itself, when a compression distortion appears over an image area that is highly textured, a threshold elevation is possible as the sensibility to the distortion is reduced in that area. This is called threshold elevation; usually this threshold elevation is modeled as a power function. In Figure 3.13, a simplified threshold elevation function is presented, where C_{T0} is the contrast detection threshold for a target as given by the CSF, C_M is the contrast of the masker, and C_T is the actual detection threshold of the target in presence of the masker.

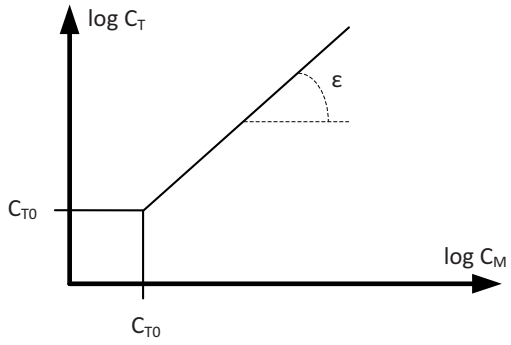


Figure 3.13: Simplified threshold elevation function

$$C_T = \begin{cases} C_{T0} & if C_M < C_{T0} \\ C_{T0}\left(\frac{C_M}{C_{T0}}\right)^{\epsilon} & otherwise \end{cases} \tag{3.25}$$

Equation 3.25 shows that when the contrast of the masker is lower than the contrast detection threshold, then no elevation is applied, but otherwise, an elevation proportional to the ratio C_M/C_{T0} is applied. This dependency is plotted in a log-log graph as a straight line with slope ϵ.

Masking models assume the common understanding that the HVS perceives visual information from various frequency-orientation channels in parallel. Models differ in the source of the combined information. If the model considers only information from one frequency-orientation channel, it is called the intra-channel model where inter-channel or multi-channel masking models consider information from various frequency-orientation channels simultaneously.

The most common forms of masking used in encoders are self-masking and neighborhood masking. The self-masking model takes only into account for the threshold elevation the value of the frequency transformed coefficient as measure of the activity in that image position (DCT block or wavelet subband), while the neighborhood masking also takes into account the value of the surrounding coefficients.

For simplicity, most of the encoder proposals use an intra-channel contrast masking model, that, although providing lower prediction accuracy than models considering both intra- and inter-channel masking effects, have the advantage of enabling parallel processing [256], because subbands are encoded independently.

As an example of threshold elevation in DWT encoders, in the JPEG2000

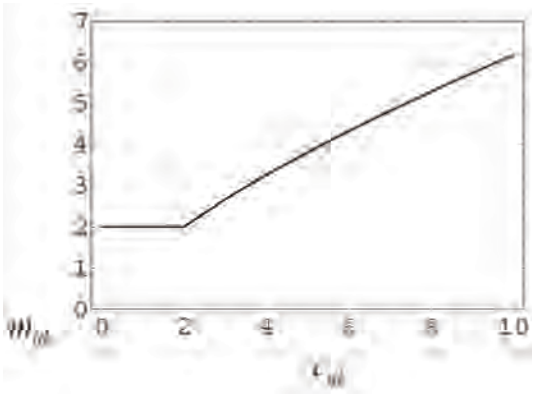


Figure 3.14: Contrast masking function

standard, a power function is applied before uniform quantization, implementing the self-masking approach. This function, with the form of Equation 3.25, is parametrized in the reference software with the parameter ϵ, which can vary from 0.6 to 1.0. For more implementation details, see [255, 242].

As an example of self-masking in a DCT encoder, in [164] Watson applied this threshold elevation in order to obtain the masked threshold m_{ijk} for each block; see Equation 3.26, where c_{ijk} is a coefficient of block k and w_{ij} an exponent that lies between 0 and 1, so for $w_{ij} = 0$, no masking is applied to that block when $w_{ij} = 1$, the threshold is constant in log or percentage terms (for $c_{ijk} > t_{ijk}$, as exposed previously a value of $w_{ij} = 0.7$ is commonly used). Figure 3.14shows the contrast masking function for $t_{ijk} = 2$ and $w_{ij} = 0.7$ [164].

$$m_{ijk} = Max\left(t_{ijk}, \left|c_{ijk}\right|^{w_{ij}} t_{ijk}^{1-w_{ij}}\right) \tag{3.26}$$

3.2.3 Perceptual coding approaches

In this section we will review some of the most relevant works that include any of the aforementioned perceptual techniques in their coding proposals.

- One of the first works that includes perceptually adaptive strategies to encode natural images is [246] where authors design an optimum quantizer based on the Laplacian PDF. The dynamic range of the coefficients, obtained from a block (16x16) based on cosine transform, was adjusted with a multiplicative range scale factor obtained from the rate control stage and modified by a

masking function. The masking function is computed and applied to each block, based on an activity index that is derived from the sum of the block coefficients. The authors also introduce a weighting factor that multiplies the cosine transformed coefficient. These weights are based on the CSF model proposed by [240] that has been adapted and normalized to have a peak at 3 cpd of visual angle. As the HVS weighting factor does not affect the PDF of the quantized coefficients, they apply a Laplacian quantizer based on the Max quantizer algorithm [231] with a deadzone for quantized values lower than 0.5.

Block based transforms are sensitive to coarse quantization when the dynamic range of the coefficient of adjacent blocks is different, because of the loss of correlation of quantized coefficients of these blocks that finally produces the blocking effect. In [246], authors include a re-quantization stage that tries to minimize the loss of correlation between adjacent blocks. They supervise the distortion between adjacent blocks to preserve it below a fixed threshold.

When the first works present their results or compare with others, they normally use the PSNR or R/D plots where PSNR is the distortion metric. As we will see, most of the latest works are still using PSNR in their comparisons, and only few of them include QAMs, like MS-SSIM, SSIM, or VQM, in spite of the advances in this field, in their performance comparisons.

- In [235], the authors proposed the Perceptual Image Codec (PIC) encoder, where the results are presented as printed images for different compression ratios able to distinguish the benefits of the perceptual proposal. The authors apply a 4 level and 4 subband frequency decomposition of the image via a Generalized Quadrature Median Filter (GQMF).

  A texture masking model is also applied. The model, which defines a perceptual quantization strategy applied to each coefficient, is based on the results of several subjective tests, performed for each image and subband, to obtain a contrast base sensitivity and a base brightness sensitivity. These base values are later modified with the inclusion of a texture masking stage, that as in [246], is based on an energy measure, but in this case on the subbands and in the pixel domain. The aim is to preserve the most sensitive information from the quantization step. This schema, supported with subjective tests, will be repeated in several proposals.

A common way of adaptive thresholding used by many encoders is to make the coefficient thresholding levels inversely proportional to the sensitivities to the corresponding spatial frequencies given by the CSF.

Coefficients corresponding to less insensitive frequencies will be more harshly thresholded than those corresponding to frequencies of higher sensitivity. Differences among encoders that use the CSF lie in the CSF model being used.

- In [257] the authors include most of the of the perceptual strategies used in perceptual coding. Although these strategies will be further refined in posterior works, and even reformulated, this work illustrates very well the aim of perceptual coding. The proposed strategies are DCT based but they can be adapted to the DWT schemas as well .

The removal of subjective redundancy is an irreversible process and involves discarding information that is not supposed to modify the perceptual quality noticed by a human observer.

- The removal of subjective information from the transformed DCT coefficients is managed in [257], as two separate psychovisually guided quantization stages, thresholding and quantization stages. For the thresholding stage, they use the CSF model proposed by [246] which provides a NxN matrix of sensitivities values. But they normalize it by averaging with the power at each frequency. This averaged power is empirically calculated from a set of images. They also include luminance masking in the calculation of their proposed NxN sensitivity matrix, which must be uniformly scaled to obtain the final threshold level for each frequency. They perform a set of subjective tests for a fixed viewing distance in order to determine the scaling parameter and the threshold values for lower frequencies that avoids the blocking effect.

  For the quantization stage, once thresholded, the quantization steps applied to each block coefficient are uniform, but the optimum step size for each coefficient depends on the threshold value for the frequency that this coefficient represents. The step size for each block also depends on the spatial activity of the image in the block region, with those blocks located in high spatial activity areas having larger step sizes . The activity of a block is determined by the masking function, which is also subjectively tuned so that the finally obtained step sizes produce sub-threshold distortion. They propose a masking function that is based on a Laplacian edge detector in the spatial domain but transformed to operate in the frequency domain.

  They compare the performance of their perceptual strategies with the standard DCT compression results, for images and video sequences, providing the compression results in bit per pixels and the bit rate in Mb/s. Additionally, in the paper, some images are exposed for visual comparison, but no quality metric has been used, not even PSNR.

- In [164, 258, 167], Watson proposed an important approach named DCTune for visual optimization of DCT-based compression schemas that is clearly exposed and covers most of the strategies followed in perceptual coding proposals. The proposal is adapted for individual images and uses luminance and contrast masking to generate adapted quantization matrices that should be sent to the decoder.

  Watson uses initially, the [237] measurements of threshold amplitudes for DCT basis functions. He then modifies these thresholds with the inclusion of Luminance masking, producing a luminance masked matrix. Each of the luminance masked thresholds is calculated either with the formula proposed by [250], or by an approximation with a power function. Then a contrast masked matrix is further computed. This matrix is computed for each coefficient following the formula given by [254, 259] but taking into account the previously luminance masked threshold. The contrast masked matrix has then a *Masked Threshold* value of Mt_{ijk} for each DCT coefficient in each block that includes luminance and contrast masking. Then, the just noticeable distortion is calculated for each coefficient in each block as the value of ϵ_{ijk}/Mt_{ijk}, where ϵ_{ijk} is the quantization error as in Equation 3.6. By pooling via a Minkowski metric through all blocks in the image, a just noticeable map for each DCT frequency is obtained that is further used to obtain a single perceptual quality value obtained for that image. This is used to optimize the perceptual quality for a desired bit rate and vice versa.

  The main drawbacks of [164, 167, 258, 260] are that the proposals are defined for individual images, fixed viewing conditions, and only for gray scaled images. It is not locally adaptive enough; all 64 transform coefficients share one common texture correction factor. Regarding the results, they are presented as printed images for visual comparison without giving at least PSNR values or presenting R/D plots comparing with other proposals, but the perceptual gain is clearly shown by inspecting those images.

- In [251], authors extended and generalized their previous works to account for color images, variable and parametrized viewing conditions and making the model image independent. The authors present a model for predicting the visibility thresholds for DCT coefficient quantization errors from which a quantization matrix design method is proposed. The model is parametrized based on experimentally measured visibility thresholds as in previous works.

A variety of models has been proposed for the evaluation of image quality and image fidelity that are usually based on a set of oriented filters [162, 161, 115, 163] see images 2.23, 2.24, and 2.25.

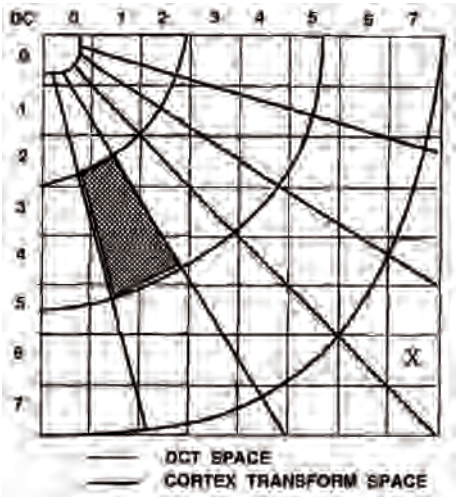


Figure 3.15: DCT-Cortex Filters Mapping

The main problem of the application of the oriented filter-based models to DCT image coding is the required conversion of the model domain to the DCT domain. Application of the filter bank HVS models would be advantageous, because this would allow the use of the state-of-the-art HVS models for DCT image coding. In [261], a detailed explanation of how the mapping between the Cortex Filters and the DCT transform is made. In [188] a general method to combine models of orientation filters with the DCT transform domain (it can be extended to other transforms too) is proposed by calculating a local sensitivity factor for each DCT (color) block.

- The authors in [261, 262] propose another adaptive perceptual masking threshold model for image compression. This model can be used for the *thresholding stage*, as proposed by [257], where separation of quantization in two stages was justified: the *thresholding stage*, for fixing the deadzone, and the design of quantization steps in the *quantization stage*. The proposed model is in turn separated in two stages. The first one, based on [251], provides a quantization matrix that is dependent on the viewing conditions but image independent. In the second stage, by the inclusion of an estimation of the texture energy, an image dependent matrix is given. This is also an interesting approach because a mapping of the DCT coefficients onto the Cortex filters to estimate that texture energy is done. In Figure 3.15 from [262], this mapping can be seen. They define a set of *overlaping matrices* where each of the 8x8 factors of the matrix contains the approximate energy contributed by the corresponding DCT coefficient into the cortex band for that overlap matrix. In order to get the final threshold

value, they first compute the energy contained in each cortex band, then after an threshold elevation mapping function, the threshold value is obtained. The threshold value obtained, Mt_{ijk}, with this proposal is more accurate with the HVS but also more aggressive than in previous proposals. For presenting their results they use printed images at different rates for visual comparison jointly with bit rate savings graphs.

- In [188], the filter bank used is a variant of the Steerable Pyramid proposed by Simoncelli et al. in [163]. They propose the WMSE distortion measure, commented on in Section 2.5.2. They translate each of the basis DCT functions into the filtered domain, obtaining a Weighted sensitivity for each DCT coefficient. The results were presented for one unique printed image for visual inspection with no other type of PSNR tables, R/D comparison curves, etc.; nevertheless the quality gain is clearly appreciable.

- Quantization error produced in DWT based encoders has been studied by Watson et al. in [263, 106], where a model for DWT noise detection thresholds is presented. These models is a function on the level and orientation of the wavelets subbands and also on the display resolution. With this model, a perceptually lossless quantization matrix can be calculated and adaptive quantization schemes could be developed. With this matrix, all errors should be below threshold. The authors follow the methodology and strategies used in previous works with the DCT transform [251, 260, 264], adapting to the wavelet transform as they detail in [263]. The main problem is basically the same, to find the error visibility in each wavelet subband instead of in each DCT coefficient (that represents a frequency range).

  In order to determine the visibility of the wavelet quantization noise of each wavelet subband, they perform subjective tests with different stimuli located in different levels, orientations and spatial resolutions. They provide a mathematical model for the basis function amplitudes for each level and orientation for a six-level Antonini 9-7 DWT. They also provide a final expression that uses the display resolution to get the quantization matrix. To present the performance results of the proposed method, they provide only two images compressed with the perceptually proposed lossless DWT quantization matrix and twice that matrix, for visual inspection. For the one compressed with the lossless matrix quantization errors should remain invisible at the correct viewing distance (24 inches aprox.)

As shown, quantization matrices have been also used in the DWT domain, so each component of the matrix represents one subband of the DWT transform that corresponds to a level and orientation. There are 4 possible orientations

indexed as {1,2,3,4} = {LL,HL,HH,LH}, where L and H represent low-pass and high-pass filtering, respectively. Typically, a uniform quantizer (see Section 3.1) is used for each DWT subband.

- In [265] Wu and Gersho proposed a recursive algorithm for generating quantization matrices for the JPEG standard. Their algorithm begins with a coarse quantization table that recursively, coefficient by coefficient, is refined until the ratio of decrease in distortion to increase in bit rate is approximately maximized. Based on that work, [266] simplified the recursive algorithm and introduces the CSF function proposed by [247] to obtain a weighted quantization matrix for the JPEG encoder. The authors also used a CSF function [267], derived form the Campbell and Robson CSF, obtained via sine wave gratings, so that another weighted quantization matrix is obtained. Results from the original Wu-Gersho algorithm and the ones obtained with both matrices and the JPEG standard were compared. They presented the results with R/D curves, for their best CSF based algorithm, against the original Wu and Gersho algorithm and the original JPEG. The curves showed that when measuring distortion with the PSNR, the Wu-Gersho algorithm and their proposal perform likewise, outperforming the original JPEG. But when they show several pictures for visual inspection, then the best visual quality is obtained by the algorithm that implements the [268] CSF (the one obtained from sine wave gratings).

- In [269], the Adaptive Picture Image Coding (APIC) was presented, based on [235] but including two major advantages, adaptivity and intra-band masking. In [235], the quantizer levels for each subband were selected based on the pixels having the minimum available amount of masking and they should be transmitted to the decoder. In this proposal, the quantizer levels were obtained taking into account the amount of masking present in all the pixels for the current subband. They exploit the smoothness of the amount of masking in natural images, using the amount of masking found in the previously encoded pixels as input to a masking predictor for the rest. So, the masking predictor uses the masking information of the neighborhood (only already encoded pixels).

  Id addition, the way they encoded the quantized coefficients, using known maps at the decoder, avoids the need to send quantization steps as side information. As in [235] and previous proposals, the local noise tolerance is based on a detection threshold multiplied by a masking adjustment factor. The detection threshold includes luminance masking that has been subjectively obtained for each band. The masking adjustment factor accounts for contrast masking but in this case the authors introduce the masking produced by large image components located in the same position

but in other subbands. Results of APIC widely outperforms the PIC ones in rate savings and in perceived quality. They present results as printed images with approximately the same perceptual quality but at different rates. Bit rate savings for the same perceptual quality arise up to 40%.

One of the major questions that arise when facing this way of presenting the results is how the perceived quality is measured.

In most of the works, no QAM was used to measure bit rate savings to the same objectively measured perceptual quality.

As exposed previously, two stages are commonly used to introduce perceptual quantization:

1. The first is to fix the deadzone size via the quantization threshold, normally based on CSF that may or may not include luminance masking.

2. After that, in a second stage, the quantization step sizes are also tuned by using the CSF and usually an adaptive quantization is performed, tuning the quantization steps further, to account for contrast (or texture) and luminance masking.

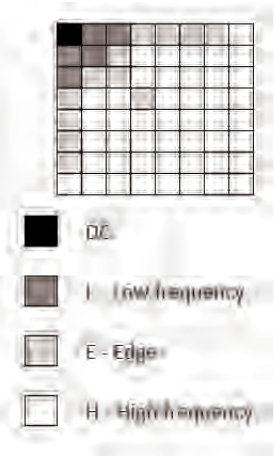


Figure 3.16: Tong block coefficient clustering.

- In [253], Tong presented an interesting approximation to that second stage for a JPEG perceptually enhanced encoder. Once the quantization matrices have been thresholded, they use the baseline JPEG proposed matrices. Their

proposal is an adaptive way to modify those initial quantization values by scaling them, for each of the k DCT blocks, with a factor $m(k)$ composed by two components, $TexMask(k)$ and $LumMask(k)$, that account for contrast masking and luminance masking, respectively. They propose a classification of the DCT blocks in three classes, texture, edge and plain blocks. They clustered the block coefficients in the three regions. The sum of the energy of each region, see Figure 3.16, is used to determine, through an algorithm that uses empirically obtained energy thresholds, to which of the three classes that block belongs. Based on that block classification, an adaptive calculation of the $TexMask(k)$ and $LumMask(k)$ factors is provided. Multiplying these factors, the $m(k)$ scaling matrix is obtained. The matrix that finally is used to quantize that block is obtained by scaling the initial quantization values with the scaling matrix. Results are presented as percentage of bit rate savings. So, for that mentioned second stage, the inclusion of this adaptive scaling matrix over the baseline JPEG one is reported to get bit rate savings from 5% up to 22% for the same perceptual quality. In order to determine the perceptual correspondence of compared images, the authors performed subjective tests. The model requires a computational overhead of 10% only on the encoder side.

- In [270, 271], Taubman presented the EBCOT algorithm for embedded bit-streams (Embedded Block Coding with Optimized Truncation of the embedded bit-streams) that was finally adopted as compression framework for the JPEG2000 standard. In [271], he reported PSNR results for various images and bit rates, i.e., obtained with a MSE as distortion metric for the Post Compression Rate-Distortion Optimization (PCRD-opt) algorithm. He presented also a new spatially varying visual distortion metric to be used in the PCRD-opt algorithm, that was lately tuned to be included in the standard JPEG2000 Extensions Part II.

  An important decision in [271] was to fix the *visibility floor* to a single small value for all subbands. This is important because on that basis, the distortion metric can be calculated independently of any assumptions on the viewing distance, which is highly desirable for uncontrolled viewing conditions, i.e., assuming the worst case.

- As he states, in previous works [258, 164, 167, 269, 272] visual masking effects were taken into account by explicitly modifying the quantization parameters, therefore, scalable compression was not considered and rate-control must be performed iteratively. His visual masking distortion metric is closely related to the one used in [164, 269] so that the formula parameters are set to the same values. It is adjusted by a *visibility floor* term (masking JND in other works) that establishes the visual significance of the

visual distortion in the absence of masking and by a *visual masking strength* operator that accounts for masking strength based on the masking values in the current coefficient's neighborhood. The neighborhoods are calculated for each 8x8 cell in which each code-block is divided using the same masking strength value for all samples in any given cell. The proposed algorithm is implemented in JPEG2000 VM3 and compared to SIPHT. The author provides data tables with PSNR values for some of the most popular images at different bit rates, and his proposal obtains better results in PSNR. But also for 2K images, some cropped regions are printed for visual inspection, and comparison, at the same rate the proposed technique provides much better perceptually quality than SPIHT, and with the equivalent perceptual quality the proposed method is able to encode with 0.2 bpp less bit rate.

- In [273], Zeng et al. presented an important contribution that was included in the standard JPEG2000 Extensions Part II, and it is widely explained [255, 274, 275]. They presented an improved visual masking function that includes the benefits of the self-masking and the neighborhood masking strategies, avoiding some of the issues with these techniques [255]. They proposed the application of the weights given by the formula into the post compression rate-distortion optimization (PCRD-opt) algorithm to control the amount of masking of each code-block. This is the same strategy used to apply the CSF in the PCRD-opt algorithm, as exposed in [255]. The proposed masking approach non linearly maps the wavelet coefficients to a perceptually uniform domain prior to quantization. It is essentially a coefficient-wise adaptive quantization. As it is included in the PCRD-opt algorithm, it allows bit-stream scalability, as opposed to many previous works. Some figures were presented to differentiate the effects of the three masking procedures, and it is mentioned that for images, like woman, a bit rate savings up to a 50% can be obtained with the point-wise strategy.

As shown so far, most of the works include values of visual contrast thresholds in their perceptually adaptive quantization, chosen either from any of the models of the CSF or obtained by a set of subjective and psychovisual experiments.

Most of these works have been oriented to obtain a visually lossless or sub-threshold compression and then compression becomes lossy by scaling the values of the quantization steps, i.e., assuming that the relationships between quantization values of sub-threshold matrices hold for supra-threshold compression.

- In [276], Hemami and Ramos performed a series of psychophysical

experiments in order to determine if a uniform scaling of quantizer steps sizes in DWT based schemes is valid for supra-threshold compression. After that, in [277] they propose a non uniform scaling strategy for supra-threshold compression. The psychophysical experiment was designed to analyze the spatial masking in natural images and how different quantizer step sizes at different levels and orientations affect the visibility of artifacts. They found a strong correlation between the Minimum Noticeable Quantizer Step Size (MNDSS) that produces a minimum noticeable distortion and the subband standard deviation, indicating the presence of orientation-dependent spatial masking. So, the quantizer step sizes were parametrized in terms of subband standard deviation. They also found that the contrast sensitivity was higher at subbands with higher energy, so for these subbands masking thresholds should be lower. And the content (edge/detail) influenced also the value the (MNDSS). They provide the formulas and the value of the parameters, obtained from their subjective tests. These quantization step sizes can be directly used in non embedded wavelet encoders, but also they propose the formulas that provide the weights that can be used in embedded wavelet encoders.

They use a simple intra-band subband coder with run-length Huffman coding, to compare with the Watson DWT proposal [106] where the sub-threshold quantization matrices were simply scaled for supra-threshold compression. They provide several images for visual inspection and also a table with percentages of bit rate savings for six common images. Bit rate savings were in the range from 17% to 22% when compared to [106] at 0.15 bpp, having the same perceptual quality. The authors do not mention how this perceptual equivalence was measured. They also compare their results against the SPIHT embedded encoder, with and without their proposed weights. For this comparison, they only present two images at 0.2 bpp for visual inspection, using a 4 level DWT.

These formulas, which link for each subband the minimum quantization step sizes and the standard deviation of the subband coefficients, are interesting because only with this statistical information can a better perceptual supra-threshold compression be achieved.

As the quantization step sizes, obtained this way, are specifically designed for supra-threshold visually noticeable distortions, for a sub-threshold compression any of the previous approaches can be used and then, when further compression is needed, his proposal can be employed. One minor problem then is to identify the at-threshold compression level when we have to change the strategy. Therefore, for sub-threshold compression, the MNDSS values can be used and then a uniform quantization can be applied to obtain the desired bit rate.

- In [278], Wang et al. proposed and clearly explain the Daly method to apply the CSF to the DCT coefficient taking into account visual parameters as the dot pitch of the monitor and the visual distance, as proposed by Watson in [106]. The method in [278] was also used in [279] for visual modulation of halftone patterns, and was initially proposed by Daly in [280]. The Daly model is based on the Mannos and Sakrison [240] CSF proposal. The authors use the PVRG-JPEG Codec [281] with their quantization matrix instead of the baseline JPEG proposed matrix. Results were also compared with several JPEG encoders; a EZDCT encoder [282], a DCT based embedded encoder, an Adaptive Thresholding JPEG coder [283]and Joint Optimization JPEG coder [284]. The results were presented in a table for the Lena and Barbara images compressed at different bit rates and all compared encoders, with PSNR the quality metric used. The results show that with the simple incorporation of the proposed HVS quantization table for DCT encoders, the performance in PSNR is almost the same as that obtained with much more complex encoders, like EZDCT, which in their comparisons is the best performing one; it provides similar PSNR performance as Shapiro's EZW coder.

As shown, the simple use of contrast sensitivity thresholds applied to each subband is a good approximation for sub-threshold compression, but as many authors state, this does not guaranty the best perceptual performance for supra-threshold compression.

Therefore, some authors have proposed adaptive modification to account for supra-threshold compression and therefore perceptually driven rate-control mechanisms appear to obtain the best perceptual performance in supra-threshold compression.

- Höntsch and Karam in [272] propose a DCT based encoder that includes a perceptual adaptive rate-distortion mechanism that is able to reduce to the minimum rate possible a DCT encoded image while preserving the maximum (at-threshold) perceptual quality. The algorithm is based first on a CSF threshold adjustment followed by a contrast masking adaptive adjustment. Furthermore, the algorithm estimates the masking adjustment based on the already quantized coefficients; therefore, this adjustment can be calculated again in the decoder side that avoids the need for sending threshold information to the decoder. Therefore, the gain in rate obtained by the reduction of perceptual redundant information is not wasted in sending side information. They propose an error metric that determines the amount of error at a specific DCT band and if this error is above thresholds. For each band, a quantizer scaling weight is iteratively obtained. These weights

must not be transmitted because in the decoder they can be obtained again by comparing the quantized coefficients with the threshold model. The contrast sensitivity thresholds are calculated with the model proposed by Ahumada and Peterson [250]. The authors compare their results with the DCTune Watson proposal, providing a table with the bpp needed to encode different images with both methods, and R/D plots for two of the images where the distortion metric is PSNR, as well as printed images for visual inspection. Unlike previous works, the authors perform a set of subjective tests in order to perceptually compare their results. The subjective test results, exposed as MOS values, state that for low bit rates, the proposed method provides better results than DCTune.

- In [252], Karam and Watson introduce their previous research into the JPEG2000 encoder to improve it with a perceptual distortion control, i.e., instead of encoding for a target rate, the target is a desired perceptual quality. They include the work in [106, 272], the perceptual model and the distortion rate control, into the JPEG2000 standard (using the JASPER2000 implementation) so that is fully compatible with Part I, and the perceptual optimization included in the Extensions in Part II of the standard have already been replaced in their proposal.

  They compare two versions for their proposal, the *approximate distortion control*, where only the CSF thresholds have been included for each subband, and the *precise distortion control*, where in addition the masking and luminance thresholds are adaptively used. As performance comparison, they provide R/D curves (with PSNR as distortion metric), showing the behavior of both versions versus the conventional coding. The R/D performance of the *approximate* version is almost the same as the conventional, but the R/D performance for the *precise* version is approximately 2dB worse. This fact is justified. For the *approximate* version, the perceptual model equals the performance of the Part II perceptual extensions included in the standard. The *precise* version is designed with the perceptual error metric and not with the MSE, but the R/D comparison is still made with PSNR as distortion metric. Therefore, they also presented printing results for visual inspection where, for the same rate, images from the conventional coding and the *precise* version are perceptually indistinguishable, the latter one having lower bit rate. The bit rate savings were in the range of 14% to 20%. They also perform comparisons with the inclusion or not of the neighborhood masking strategy, obtaining for the version with neighborhood masking slightly larger savings (15% to 21%). The perceptual benefits for the inclusion of the neighborhood masking do not justify the increased complexity for that version, so they finally adopted the self masking strategy.

When facing high resolution images (above 2K), the proposed *precise* version clearly outperforms the conventional one. A set of subjective impairment tests have been conducted to compare the coding performance in terms of perceived quality for the proposed JPEG2000 encoding with precise perceptual distortion control and the conventional JPEG2000 encoding. The results are given as R/D plots where the distortion metric is MOS. The plots show that using MOS, now the *precise* version is above the conventional one for low bit rates.

- Extensive experiments run by Nadenau et al., and described in [285, 249] allowed the introduction of an adaptive way of performing a complete adjustment of the CSF shape to the coefficients in each of the DWT subbands, via Finite Impulse Response (FIR) filters. Those filters model the CSF in each subband. In [249], the authors include their proposal in the JPEG2000 encoder and also comprehensively review the non adaptive schemes used in the literature to apply the CSF. Their FIR based proposal could be applied also in two ways, in the quantization stage called Adaptive Coefficient Modification (ACM), or as part of the PCDR-opt process called Adaptive Modification Distortion function (AMD). If the ACM solution is chosen, the inverse FIR filters must be applied at the decoder which, as they stated, could produce saturation problems. For both solutions, the FIR filter must first be designed (that design is CSF model dependent), and must be applied to each of the subbands, including then computational overhead. If the ADM is chosen, then the FIR filters must be applied to a copy of the DWT coefficients in order to be able to construct the weights for each code-block based on the weights for each coefficient in the block. This produces memory overhead that could be important for huge images, but avoids the saturation problems of the ACM solution, the FIR filter length can be reduced because the analysis/synthesis error produced with a smaller filter size can be compensated with the PCDR-opt process and also avoids the need to run the filters at the decoder. The proposed FIR filtering technique produces good results at-threshold compression, but only the ADM solution can be easily extended to supra-threshold compression by means of the PCDR-opt. Nevertheless, the proposed solution can be further improved by including some of the contrast masking adaptive proposals.

  The ACM proposal was tested in JPEG2000 and compared via subjective tests using printed images at 267 dpi. In these tests, viewers were asked to separate several compressed images (at different rates) into three quality levels, Perfect, Good, and Refused. A compression gain could be observed only by the application of the ACM technique, with gains of 29% for the Perfect quality level and 28% for the Good compression level. The ADM proposal obtained almost the same bit rate savings as the ACM one. For

presenting the results of the ADM version, several cropped regions of the bike and woman images were printed for visual inspection. The images compare the visual quality of the plain JPEG2000 version and the JPEG2000 with the ADM version jointly with the original images. Much better perceptual quality is observed for the ADM version.

- Another contrast driven rate-control mechanism as in [272] is proposed by Chandler and Hemami in [248], but this time for DWT encoders, and the error measure that drives the rate-distortion algorithm is now the Contrast RMS Error, i.e., the contrast of the distortion errors produced in a DWT subband for a specific quantization step. The authors establish relationship among Contrast RMSE and the typical RMSE quantization error produced at each subband, and on that basis they propose a Dynamic Contrast-Based Quantization (DCQ) algorithm based on the supra-threshold contrast adaptation studies described in [134]. The DCQ first determines the relationship between this minimum Contrast JND, which is weighted contribution of the contrast JND of each subband [193]. Then it iterates until the desired bit rate is obtained. The RMSE allowed in each subband for each iteration step is obtained via the previously mentioned error relationship, estimating therefore the corresponding quantization steps. If after an iteration the actual bit rate is above the requested one, then in the next iteration another minimum contrast distortion unit is applied as a compound contribution of all subbands. The results were generated from 8 bits/pixel (bpp) grayscale original images. At high bit rates, images coded with DCQ are competitive in visual quality with those coded using PCRD with fixed visual weighting, whereas at lower bit rates, DCQ generally excels at preserving image quality by maintaining the semblance of global edge-structure. The authors presented printed images at the same rate with PSNR values as quality metric, and as expected, DCQ encoded images obtain better perceptual assessment than the PCRD ones although PSNR indicates the contrary. The authors also provide tabular results for several images where besides PSNR, another objective QAM is used, the NQM [286], based on a degradation model. For both metrics the proposed DCQ methods obtain lower quality values but again, in a visual comparison of printed images gets better perceptual assessment.

- Sreelekha et al. [287] proposed a DWT coding approach implemented in the JPEG2000, using the Ramos and Hemami [193] contrast threshold proposal for the luminance channel, but including a new model for chrominance thresholds. Unlike most of the previous works, contrast thresholds are used in both, luminance and chromatic channels simultaneously. To obtain the chrominance thresholds, they use a YCbCr color space and a series of psychovisual experiments using a similar procedure as mentioned in [272].

In the thresholding and quantization phase, first a thresholding quantization stage over all subbands is run. Then, the remaining coefficients, which are clustered following a k-mean algorithm, are iteratively quantized so that, for each subband, the quantization error remain below the detection threshold. In order to achieve a desired bit rate, they run a simple iterative procedure where in each step the thresholds are elevated. The authors provide tables with the chrominance thresholds obtained for the subbands of a five DWT decomposition for the different orientations and for some of the images taken from the live database. They provide the threshold values for the Cb and Cr color components and also the expressions and parameters to obtain for both chrominance base thresholds.

Using the base thresholds, their solution works in the sub-threshold compression range. For supra-threshold compression, they tested their proposal first by simply scaling the luminance and chrominance thresholds. The scale factors, 1.1 and 1.3 for luminance and chrominance, respectively, were obtained by trial and error, by applying them to several images and for different rates. Since the base model itself takes care of the properties of individual images as well as that of the subbands, a uniform scaling of the base threshold was enough to achieve higher compression with improved perceptual quality, compared with that of the standard codec. But as known, the luminance thresholds, which were developed for at-threshold compression, are not effective for higher compression ratios. However, it was also noticed that scaling the chrominance thresholds provides a larger tolerance and preserves the color details even at a higher scale. So the algorithm was finally tested with the luminance thresholds kept at the at-threshold level and scaling only the chrominance thresholds in order to achieve higher compression rates.

Regarding the results presentation, they use printed images for visual inspection, but most interesting is that this is one of the first works that uses some advanced QAMs for obtaining the bit rate savings for the same perceptual quality. In particular, besides PSNR, they also use the VIF, SSIM, VSNR, and also MOS values, providing the metric values for the tested images and for the two versions (luminance scaled or not). The QAMs were applied to each of the three color components R, G, and B, and the mean value of the three components is taken as the quality measure. The comparison with the JPEG2000 compressed image clearly shows superiority of the proposed algorithm in retaining the colors without losing the original shades even at much lower rates (but at-threshold). These color shade variations can be noticed in many parts of most of the test images for supra-threshold rates. The comparison between their two versions provides slightly better objective perceptual quality for the version where only

chrominance thresholds were scaled. The major problem with the proposed algorithm is the time required to converge to the given rate, as they use an iterative approach.

- Oh, Bilgin, and Marcellin, proposed a new method to determine the visual thresholds for the JPEG2000 in [288]. Although visual thresholds for the wavelet transform had been successfully employed previously, in their proposals authors include the statistical characteristics of the wavelet coefficients jointly with the dead zone quantizer.

  Previously, in [252], Liu, Karam and Watson assumed that the quantization error is uniformly distributed over the interval $(\Delta, -\Delta)$, being Δ the quantization step. But the authors provide here a PDF of a redefined model for the quantization distortions produced by a dead zone quantizer and mid-point reconstruction. Using this model, which depends on the size of Δ and on σ, the standard deviation of the subband coefficients, they performed a series of subjective experiments in the same way as in [276]. For a uniform gray (128 gray level) image that is transformed with the DWT, they introduced in a single subband (the rest of subbands remain unaltered) a patch of coefficients altered with the new quantization error model. As the error model depends on Δ and σ, then for a very low fixed Δ, only by modifying the σ in the patch they obtained several patches with increasing error. They performed the subjective test in order to model the visual thresholds corresponding to the new error model, and finally, the authors provide the equation that relates the standard deviation σ to the visual threshold. For the results presentation, only a table with bit rate reduction for the performed comparison is provided. With this new approach included in the Kakadu v6.1 (JPEG2000 compliant encoder), the authors obtain an additional 30% of bit rate reduction for the same perceptual quality than in [289] for the same digitized radiographs.

  The relationship between the visual thresholds for the sub-threshold level and the standard deviation of the subband coefficients in wavelet based encoders with dead zone quantization is an important finding.

- Going a step forward, the same authors include visual masking adaptation to their previous quantization error model in [256]. The previously obtained Visual Thresholds (VT) were adjusted further using visual masking effects present in the background image. A masking threshold value is used to elevate the VTs. This masking threshold value is composed of two weighted components to account for self masking and texture masking. Compared with numerically lossless compression of JPEG2000, the proposed method achieves a 60% bit rate reduction on average, without visual quality degradation. The bit rate reduction obtained when compared to their

previous version in [288] is on average 40%. Furthermore, the proposed method yields superior image quality at equivalent bit rates, compared with conventional JPEG2000 encoders. The results are presented in tables of bit rate reduction for several commonly used images and with some cropped regions of 2k images where the quality differences are clearly perceived. In [290], authors extend their previous models to chrominance, providing the Δ values for the Cb, and Cr channels used in their model. In this case, they compare their results in bit rate savings against the visually lossless proposal by Chandler et al. in [289], obtaining an average savings of 5.50%.

3.2.4 How proposals compare their results

One of the first ways of presenting how good a specific perceptual encoding proposal works was to use printed images for visual inspection. This is a good approximation if the new proposal exhibits clearly higher performance with the compared methods and the benefits are noticed this way. Normally, several tables with PSNR values as quality metric were also presented. Those values and images are normally presented only for a reduced set of specific rates or even only for one rate.

Another common strategy for presenting results is by the way of bit rate saving values, normally in tables for a reduced set of qualities. Bit rate savings are supposed to occur for the same perceptual quality of the image. The main problem with this way of presenting results is how the perceptual quality is measured. Normally, authors use subjective tests or simple visual inspection to determine this equality.

Providing performance results over a wider set of bit rates by means of Rate/Distortion curves is a much better approach. With a quick look to the R/D compared curves, the reader can see which proposal is performing better. In fact, it is the most prevalent way of providing and comparing results in image and video coding research, jointly with tabled data. As reviewed before only some proposals include a QAM as distortion metric in the R/D evaluation. Most of the works still use the PSNR as distortion metric when using R/D curves because in general it is considered that PSNR is well correlated with perceived quality, as long as the saturation limits are considered. As shown in previous chapters human perception of quality saturates above some specific quality level and for very low quality values it is difficult and it depends on the subject to determine which image is better. The use of a QAM as distortion metric in R/D is perceptually much more accurate than PSNR, but practical issues, i.e., computational cost, impose the use of PSNR.

Besides, the log scale of the PSNR metric makes comparing numerically the gain of quality for two points in a curve difficult, i.e., the comparison of a specific gain in dB at high bit rates with the same gain at low bit rates has different perceptual meaning. In addition, although presenting a R/D curve is easy and provides a quick way to see differences between proposals, it has the disadvantage of being difficult to provide a numerical comparison value, i.e., it is easy to measure the bit rate savings for a specific quality value, or also, the quality gain for a specific bit rate, but for the whole quality range of values, or for the whole bit rate ranges covered by the curves it is difficult to provide a single measure for comparing both curves.

During the development of the video coding standard H.264/AVC, an objective coding efficiency measurement, i.e., the Bjontegaard Delta PSNR (BD-PSNR) or $\Delta PSNR$ [291, 2], was proposed. Many image and specially video encoder proposals use it now as a *de facto* standard for presenting compared results; in fact, the Bjontegaard model is used by various experts to calculate the coding efficiency of compression standards. For example, this model was used during the development of H.264/MPEG-4 AVC, the Multiview Video Coding (MVC) extension of H.264/MPEG-4 AVC [12], H.265/HEVC, and the multi-view extensions of H.265/HEVC. The Bjontegaard model is also widely used by researchers working on image and video compression to benchmark the performance of their algorithms against well-established and state-of-the-art compression algorithms. However, the Bjontegaard model might not be an accurate predictor of the true coding efficiency as this model relies on PSNR measurements. In a recent work [292], the authors use the average MOS and bit rate differences computed between the fitted R/D curves inspired in the Bjontegaard method. We will also use some modifications of this method when comparing some of our results using a QAM instead of the PSNR, therefore, we will first overview the main ideas behind this model.

The basic idea behind the method is to calculate the average difference between two R/D curves. This can be done iteratively, calculating the real obtained PSNR, for small increments of rate, so finally we obtain the average difference in PSNR. One problem, as mentioned before is the log nature of the PSNR metric so that differences in high rates have more weight in the final result. In Figure 3.17 [291], the popular way of presenting R/D curves is shown. As stated in [291], the difference between the curves is dominated by the high bit rates, so the range 1500-2000 gets 4 times more weight than the 375-500 range even if both represent a bit rate variation of 33%. This problem is resolved by using a log scale for the rate as in Figure 3.18 where the R/D curves do not deviate much from straight lines. The other problem in such a procedure is that we have to compute a large amount of results iteratively.

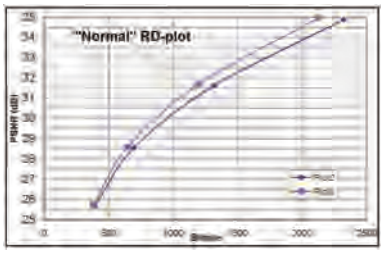


Figure 3.17: Normal R/D curves for two compared proposals

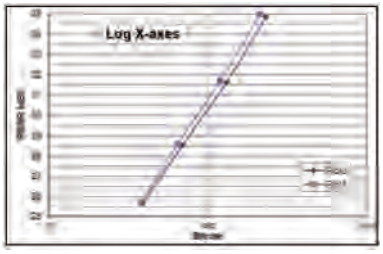


Figure 3.18: Log(Rate) R/D curves for two compared proposals

$$D_{PSNR} = D(r) = a{\cdot}r^3 + b{\cdot}r^2 + c{\cdot}r + d \tag{3.27}$$

Based on experimental observations, Gisle Bjontegaard employed a third order logarithmic polynomial to approximate a rate-distortion (R/D) curve from only four real R/D value pair, that properly covers the whole rate range and instead of using the linear rate scale, a logarithmic scale is used. So given four output bit rate points a, b, c, d, we can obtain the interpolated curve that passes through all 4 data points with Equation 3.27, where D_{PSNR} represents the reconstructed distortion in PSNR and $r = log(R)$ where R is the output bit rate.

Based on the interpolated R/D curves, the average differential PSNR between two R/D curves is calculated by Equation 3.28 where subindexes 1 and 2 denote each of the curves and r_L and r_H are the integration bounds. In the same way, the average change in rate can be obtained, see [291]

$$\Delta PSNR = \frac{\int_{r_L}^{r_H} (D_2(r) - D_1(r))dr}{r_H - r_L} \tag{3.28}$$

In Figure 3.19 [2], the area that is taken into account for calculating (1), the average PSNR gain, and (2), the average bit rate saving is shown, jointly with the integration bounds in each case.

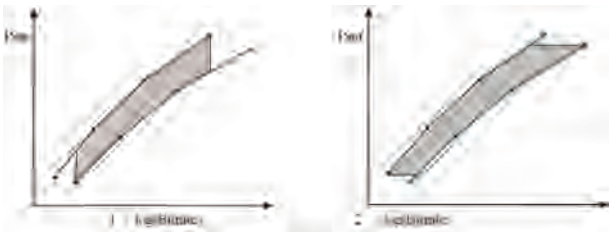


Figure 3.19: Integration area and limits for calculating 1) average PSNR gain and 2) average bit rate savings.

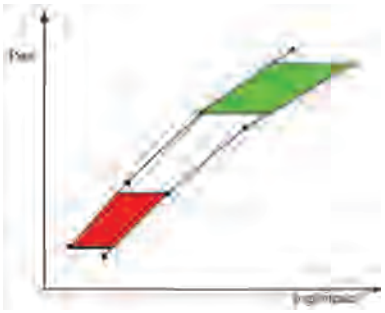


Figure 3.20: Performance at low and high bit rate.

Some refinements have been made to the Bjotegaard model; in [2], Bjontegaard shows that when the R/D curves are obtained for Qp (quantization parameter for H.264/AVC) differing in 4 or more units, i.e., long distance between curves, then providing only an average for the whole bit rate range is not enough. In this case, the use of a $log_{10}(Rate)$ scale is used and three values are given to achieve better results: 1) the average over the whole range, 2) the average of the upper section of the curves as indicated in green in Figure 3.20, and 3) the average of the lower section of the curves as indicated in red in Figure3.20. In [293], an interpolation with 5 points is proposed, and [294] presents a metric to validate if the integration area is properly overlapped.

3.3 CSF weighting matrix

As stated before, CSF is the most widely used method to include perceptual enhancements into image encoders. As reviewed in 3.2.3, most of the methods use some kind of subjective or psychovisual experiments in order to get the quantization matrix that will be included in the encoder.

In this section, we will introduce a method based on [295] to obtain these

quantization matrices directly from a model of the CSF. We will analyze the behavior of this method when used in the S-LTW encoder [3] and analyze several optimizations in order to improve its R/D performance in a perceptually enhanced version of the S-LTW.

The best working optimization has been tested with a large set of images. Since the encoder modifications are perceptually based, the results must be compared using a QAM as exposed in the previous chapter.We will use the VIF metric. Although this metric is not the fastest one, it is the one with the highest correlation with the MOS values in our tests.

In the Mannos and Sakrison CSF model [240] (see Equation 3.12) , the spatial frequency f can be expressed in pixels per degree of visual angle. As commented and explained in section 3.2.1.3, we will use half of the maximal spatial resolution of the HVS (64 cycles/degree) at a distance of 12 inches. With these parameters, the quantization matrices obtained are suited for a display resolution of 300 dpi and a visual distance of 12 inches.

The use of these parameters produces the CSF curve of Figure 3.5. The spatial frequency of the x axis must to be mapped to each level or subband in a wavelet decomposition schema. In a typical DWT schema, the input image is convolved with a low-pass and a high-pass filter for each row and then for each column. After each filtering operation, the result is down-sampled by two. This produces 4 frequency subbands named LL, LH, HL, and HH, where L corresponds to low filtered results and H to high filtered results. The first letter in each pair represents the rows and the second one the columns. This schema is repeated N times, taking the last LL subband as input for the next decomposition.

Figure 3.21 shows a representation of a 6-level DWT decomposition. In such a 2D wavelet decomposition schema, each level and subband should be mapped to its representative spatial frequency. In [249], a way to obtain this mapping is explained.

The CSF spatial frequency range is divided by two, establishing two frequency ranges. The highest frequency range corresponds to the first decomposition level and the lower frequency range will be further divided for the next levels. In Figure 3.22, the CSF curve is shown, with the x axis labeled with frequency bands for each decomposition level. So, for the first decomposition level, a representative frequency value is chosen from all the frequencies in the L1 band. A representative value is chosen for each Ln band.

Having fixed the representative frequency values for each level, Equation 3.12 give us the normalized perceptual contrast sensitivity within a range from 0 to 1. This value is directly used as the at-threshold quantization factor for

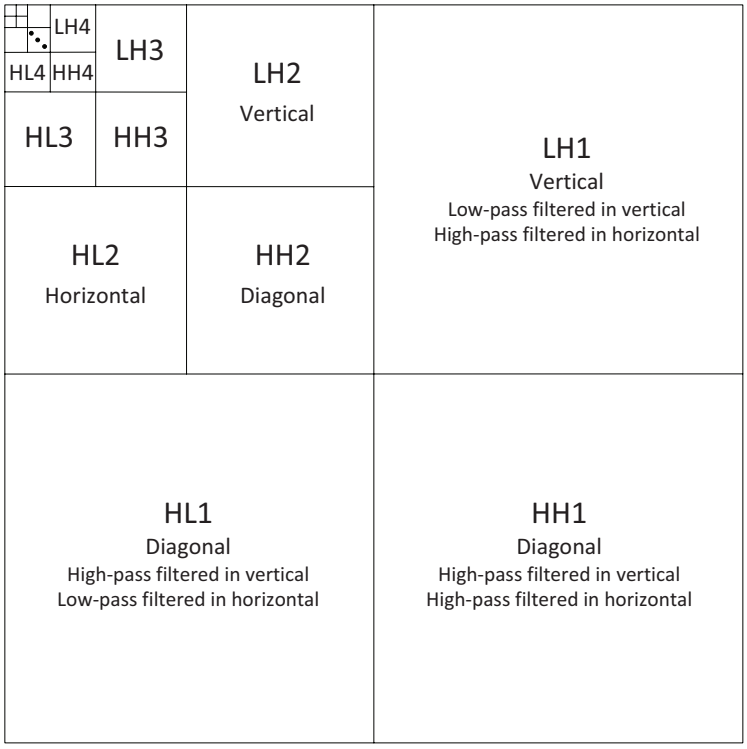


Figure 3.21: Typical DWT subband decomposition.

the corresponding DWT decomposition level. Using this strategy, i.e., with the same quantization value for each subband within a level, a CSF Quantization Matrix is obtained.

The key point in this procedure is to choose the right representative frequency value. Different sets of representative frequency values will cause different quantization matrices, which in turn lead to different perceptual qualities in the reconstructed images.

The objective is then to find the CSF Quantization Matrix that produces the image with the lowest bit rate and the highest perceptual quality, i.e., with no perceivable difference from the original. This image is told to be compressed at-threshold, i.e., further quantization will produce noticeable distortions. The perceptual quantization matrix obtained this way should exhibit the same behavior in the whole compression range, from low compression rates to high compression rates, i.e., it should maintain the best possible perceptual quality for each compression level.

As mentioned before, different ways of choosing these representative values will produce different R/D behavior. Some sets of representative values exhibit

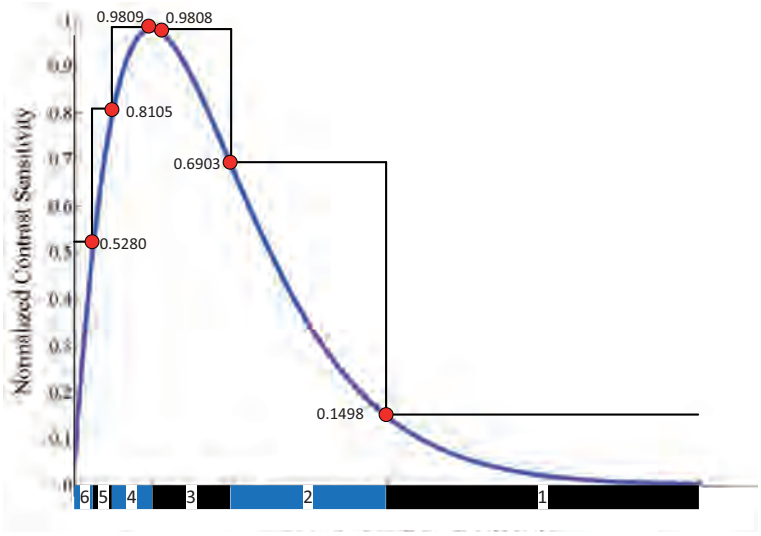


Figure 3.22: CSF curve with the frequency bands for each level labeled on the x axis. The selected representative value for each band is the peak value for each one (red points). Contrast sensitivity for these values are the quantization factors for all subbands on the same level.

good R/D behavior for low compression rates only, while others do so for high compression rates only. The objective is then to choose these representative values so that the R/D curve is maximized in the whole rate range.

We are talking about a level perceptual quantization/weighting matrix if there is only a representative value for each wavelet decomposition level, and we name subband perceptual quantization/weighting matrix if we have different representative values for each subband within a level.

In [295], the authors propose several methods to obtain the level and subband quantization matrices (called CSF Masks). The authors conclude that their level weighting matrix is the best working solution. We will name this reference proposal *bLev*. Although the authors conclude that their level proposal was the best performing one, we have also studied and compared their best subband proposal in the S-LTW encoder. We name that subband proposal as *bSub*.

We have observed that the *bLev* proposal works well in the very low compression range, although for some images and for higher compressions, as shown in Figures 3.25 and 3.24, it does not perform better than the *bSub* proposal. Figure 3.23 shows the comparison of the reference methods for the Lena image for a bit rate range up to 1.7 bpp. In this Figure, the *bLev*

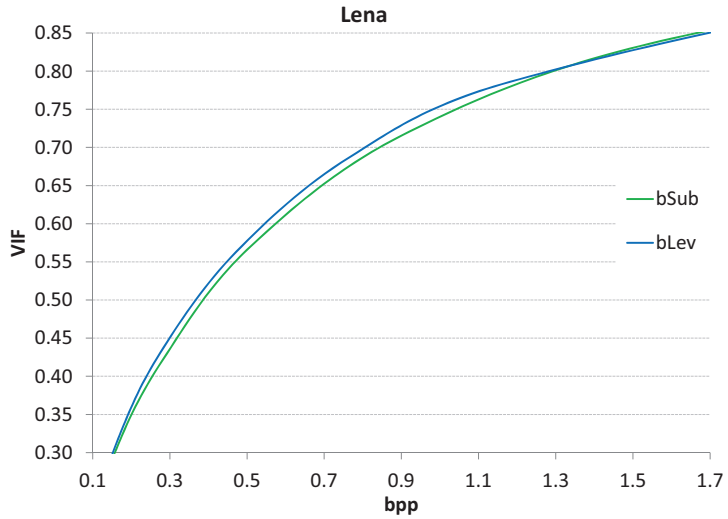


Figure 3.23: bSub vs. bLev for Lena. Comparison of the R/D behavior.

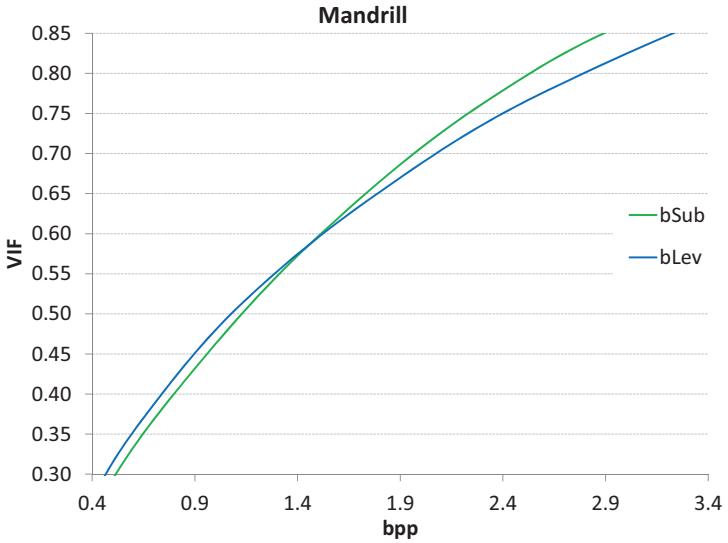


Figure 3.24: bSub vs. bLev for Mandrill. Comparison of the R/D behavior.

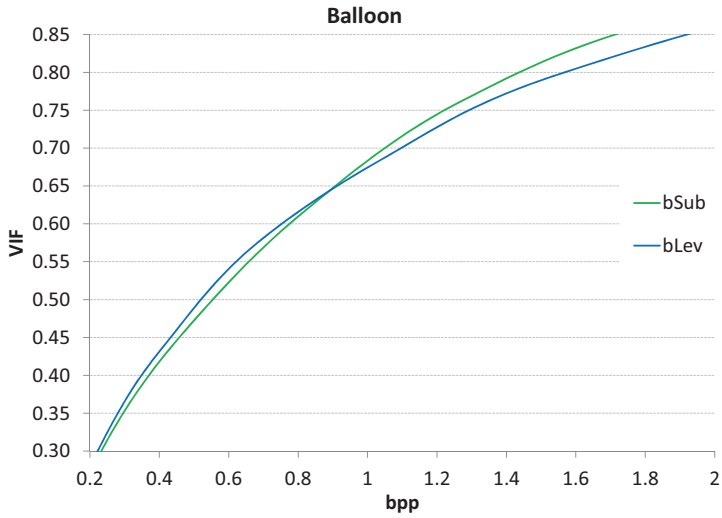


Figure 3.25: bSub vs. bLev for Balloon. Comparison of the R/D behavior.

reference method has better R/D performance than the *bSub* method. But as mentioned before, these performance differences are dependent on the image content. So, for the Mandrill and Balloon images, at figure 3.24 and 3.25, respectivelly, the *bSub* proposal is performing better in the low compression range, and the *bLev* proposal works better for higher compression.

In the *bLev* proposal, the representative value for each level frequency band is the one with the highest contrast sensitivity, i.e., the peak value of each level. The *bSub* proposal applies a 5-level DWT decomposition on the CSF curve, choosing as representative frequency for each subband the peaks of each wavelet subspace for the HH_l subbands, and $\sqrt{p_l + q_l}$ for the HL_l and LH_l, being l the DWT decomposition level, p_l the peak of the approximation subspace, and q_l the peak for the remaining subspaces of the *lth* decomposition. See [295] for more details.

We have analyzed the behavior of these two proposals in the S-LTW encoder (also a DWT based encoder) in order to see differences in R/D behavior using VIF as the distortion metric, and we also performed an analysis in order to find a perceptual weighting matrix proposal that has a better R/D behavior independently of the quality level.

When no other quantization is applied, the use of the normalized CSF values as quantization factors for each level produces, as mentioned before, the at-threshold image version, which has a reduced bit rate and a theoretical visually lossless quality.

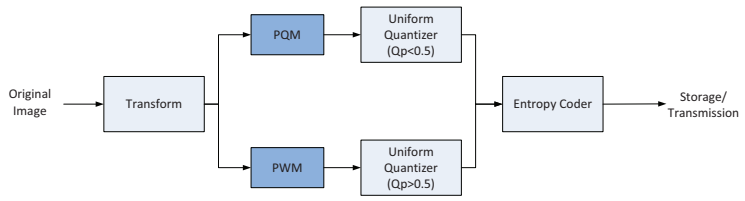


Figure 3.26: Two alternatives to obtain sub-threshold rates with S-LTW.

For most viewers and images this is true, but when both images are shown simultaneously, some viewers can detect small differences after an intensive inspection. In a normal subjective test session, where during short time periods one image is shown followed by a uniform gray image, and finally the other image is shown, these differences are unnoticeable. But when a rapid alternation of the images is done without intermediate frames, as when playing a video, and depending on the image or only some parts of it, most viewers are able to detect differences between the original image and the at-threshold version. This is because a flickering effect appears in some highly textured areas and in smooth regions of the image. These differences are however imperceptible for static images as said. This flickering effect is reduced as we move to lower compression rates, i.e., for sub-threshold qualities.

Due to this flickering effect, and because we will also test our proposals in the intra video mode with a perceptually improved version of the Motion-LTW encoder (M-LTW) [296], we need to test our encoder also at higher bit rates, i.e., for very high image qualities and very low compression.

To increase the rate for the at-threshold image, and thence the quality, two alternatives are possible. In Figure 3.26, the two alternatives are schematized.

- The first one still uses the Perceptual Quantization Matrix (PQM) with the quantization factors directly chosen from the CSF curve as explained before in the range from 0 to 1. Then, to obtain images with lower compression with the S-LTW encoder, we can use the uniform quantization parameter of the encoder to provide lower quantization. A quantization parameter of $Qp = 0.5$ does not produce any quantization with the finer quantizer, but the coarser quantizer still applies. So we can use values below 0.5 for the Qp parameter in order to produce uniform elevation of the transformed and perceptually quantized coefficients, although it must be taken into account the quantization produced by the coarser quantizer.

- The second one calculates a Perceptual Weighting Matrix (PWM) by normalizing the quantization factors, which are obtained from the CSF. This normalization can be done by dividing all values in the PQM by the lowest

one as done in the reference paper. The PQM and the PWM matrices obtained with this method, that is schematized in Figure 3.22, are shown in Table 3.1. The weights obtained this way multiply the coefficients in their corresponding levels before any quantization is done. The rationale behind this normalization is to preserve more, from the uniform quantization stage (finer quantizer) and also from the coarser quatizer, those subbands that are perceptually more important than others.

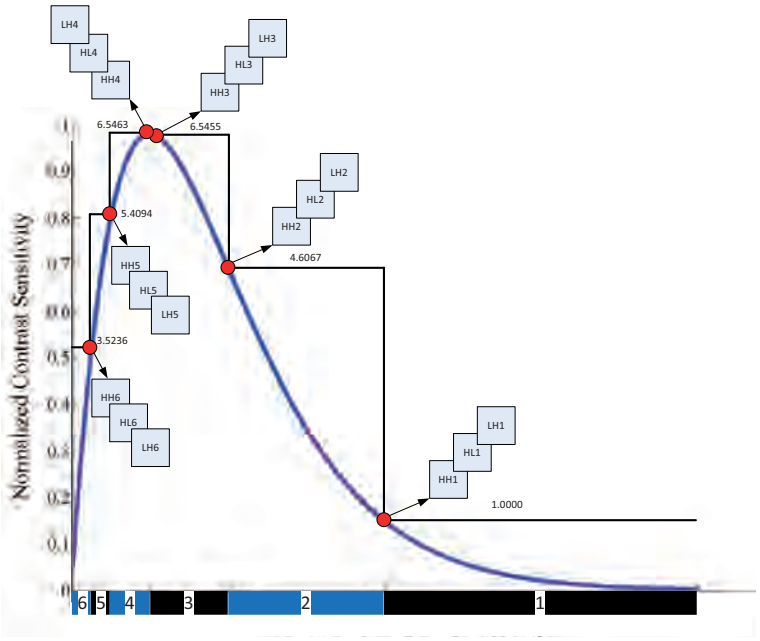


Figure 3.27: Perceptual weighting matrix values over the CSF curve with the subbands where each weight is applied.

Table 3.1: Quantization and weighting matrices for a DWT level decomposition.

	PQM	PWM
L1	0.1498	1.000
L2	0.6903	4.607
L3	0.9808	6.546
L4	0.9809	6.546
L5	0.8105	5.409
L6	0.5280	3.524

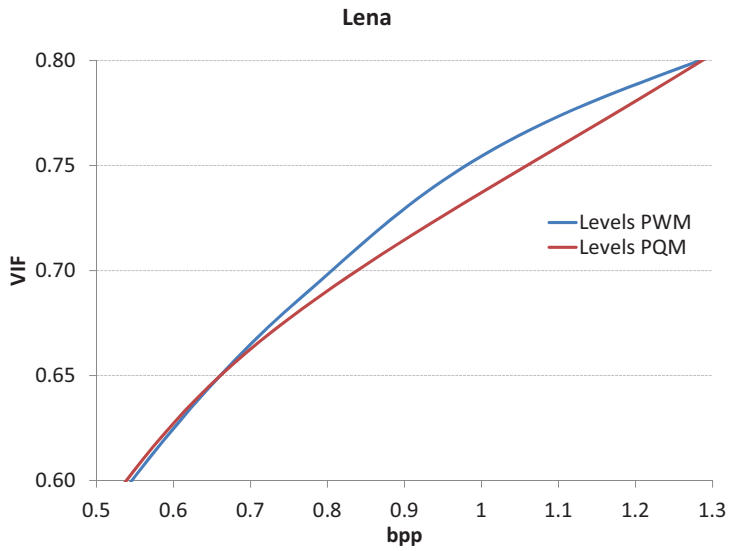


Figure 3.28: Small R/D differences between PQM or PWM plus a uniform quantization

For many images, the R/D behavior of both alternatives (using PQM or PWM) is practically the same, but for some images, slightly worst R/D results are obtained when the PQM is used. In Figure 3.28, the R/D behavior of both matrices is shown for the level decomposition. Therefore, we will use for our studies the perceptual weighting matrix (PWM) approach in order to avoid these small differences and to protect the coefficients also from the coarser quantizer of the S-LTW encoder.

The aforementioned quantization or weighting matrices are applied in a level decomposition granularity, i.e., the same value applied to the whole level. By following the same approach, we will now increase the granularity of our proposal to a subband decomposition, but improving the R/D behavior of the *bSub* reference proposal. So, we will select a representative frequency value for each subband to obtain the corresponding CSF value, and then normalize the weighting matrix.

Different R/D curves can be obtained depending on the way the normalization of the quantization factors is done, and how the appropriate representative frequency values are selected for each subband.

In figures 3.29 to 3.31, schematized images of how we assign the representative values to a subband decomposition are shown. The position of the points in these images is only an approximation to help understand the

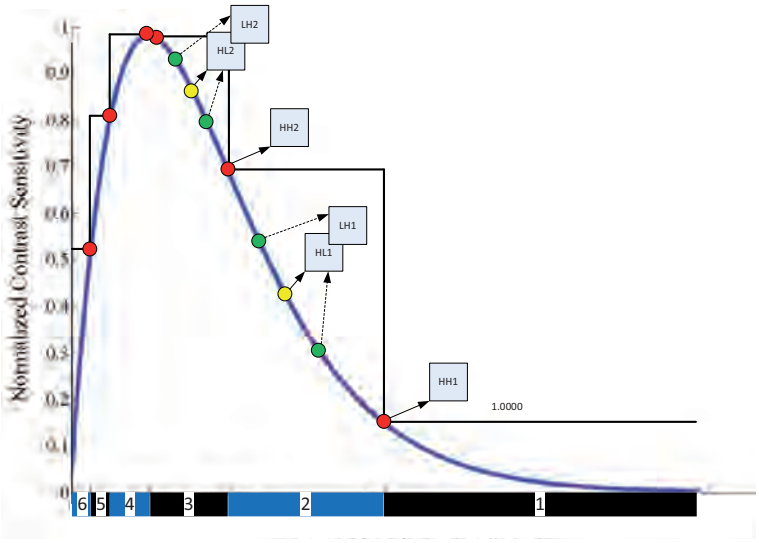


Figure 3.29: PWM-S1 and PWM-S2 representative values schema.

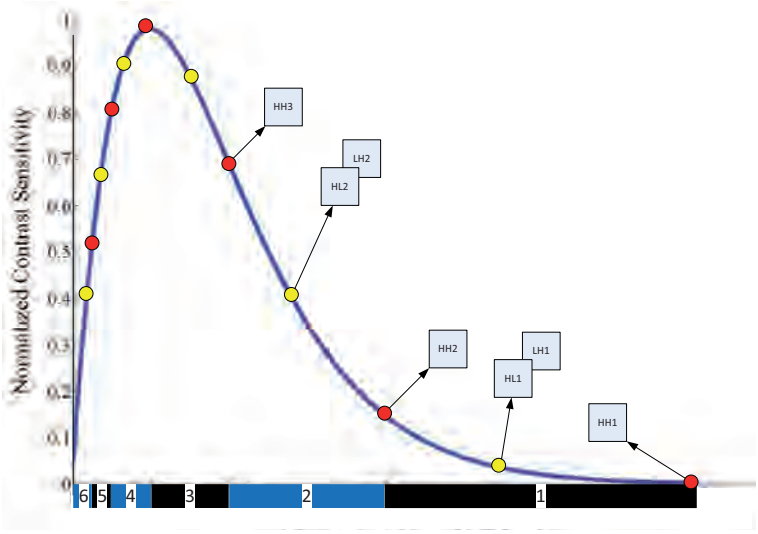


Figure 3.30: PWM-S3 representative values schema.

proposed methods. The accurate values are provided in Table 3.2.

We propose several new perceptual weighting matrices for a subband decomposition and after a performance comparison of them we will finally chose the best performing one. In all these proposals, the weights are assigned to the subbands in a high to low frequencies order: $HH1$, $HL1$, $LH1$, $HH2$, $HL2$, $LH2$, and so on. The proposals are:

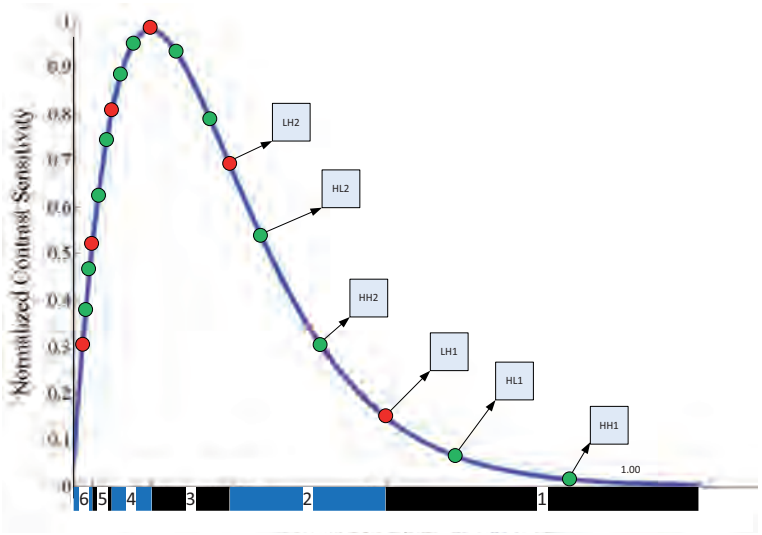


Figure 3.31: PWM-S4 representative values schema.

- *PWM-S1*: The first subband, $HH1$, get its weight from the peak value of the $L1$ decomposition level, i.e., the rightmost red point in Figure 3.29. Then, for the $HL1$ and $LH1$ subbands, the weight is obtained from the average CSF value in level 2. This schema is repeated for the rest of the levels, so for $HH2$ subband the peak value of the L2 level is chosen, and for $HL2$ and $LH2$ the average CSF value of L3 is chosen. The average CSF values correspond to the yellow points in Figure 3.29.

- *PWM-S2*: In this weighting matrix, the first subband ($HH1$) takes the same weight as in the previous proposal. For the next subbands, we calculate for each of the Ln levels ($n > 1$) the CSF values as follows. We divide the CSF curve for the Ln frequency segment in four quarters. The representative frequency points for each level Ln with $n > 1$ are the green points in Figure 3.29, which correspond to the first quartile and the third quartile (from right to left) of the curve values. So, for the $HL1$ subband the first quartile of the CSF segment for L2 is chosen. For the $LH1$ subband, the third quartile of the same segment is chosen. As in the previous proposal, this schema is repeated for the rest of the levels. When the level is located to the left of the maximum of the CSF curve, then the LH subband takes the first quartile and the HL subband takes the third quartile as representative value, while the HH subband still takes the peak value.

- *PWM-S3*: For this weighting matrix, we have shifted to the right the first representative value, the one for the $HH1$ subband. In this case, we choose the lowest CSF value. Figure 3.30 shows how the representative values have

been chosen for this proposal. With the same schema as in *PWM-S1*, the representative value for the subbands *HL* and *LH* are the average CSF values, but this time at the same decomposition level (yellow points in the figure). So for the *HLn* and *LHn* subbands, the average value of the CSF segment in *Ln* is chosen, and the value for the *HHn* subband is the lowest value of the CSF segment.

- *PWM-S4*: With the same schema as in *PWM-S2* each segment of the CSF curve for each level has been divided into four quarters, but in this case the first segment corresponds to the L1 level. Figure 3.31 shows how the representative values have been chosen for this proposal. The representative values are also the first and third quartile for each segment (from right to left), but in this case from the same segment. As in the previous proposal, when the level is located to the left of the maximum CSF value, the representative value for the *HL* and *LH* subbands are swapped, and in any case, the *HH* subband takes the peak of the segment.

In order to obtain the weights from the quantization values, we must normalize these representative values so that the lowest one is set to 1. The normalization method proposed in [295] for the level weighting matrix (dividing by the lowest value) is not suitable for all the proposed methods, as the weights become too large in the *PWM-3* and *PWM-S4* methods, and then the R/D performs worse.

The normalization that we propose for the subband weighting matrices is to set the minimum value to 1, the maximum value to the maximum value obtained for the level decomposition, and then maintain the same relative distance (on the y-axis) between the representative points that they have on the CSF curve.

In Table 3.2, the quantization values from the CSF curve for each proposal are shown for a 6-level DWT decomposition, which produces 18 subbands. In Table 3.3, the normalized values for the four proposals are shown.

In Figure 3.32, the x-axis is labeled with the 18 subband names, and each point of the curve represents the subband quantization value. In Figure 3.33, the normalized values are shown instead. As shown, the representation of the *PWM-S*4 values resembles the CSF curve more, while the *PWM-S1* proposal has up to 5 subbands with almost the maximum quantization value. Also the *PWM-S1* and *PWM-S2* proposals overprotect (elevate) the high frequency subbands.

Table 3.2: CSF values for the subband proposals

Subbands	PWM-S1	PWM-S2	PWM-S3	PWM-S4
LH6	0.4260	0.3740	0.4260	0.3183
HL6	0.4260	0.4810	0.4260	0.3740
HH6	0.5280	0.5280	0.5280	0.4810
LH5	0.6900	0.6200	0.6840	0.5298
HL5	0.6900	0.7600	0.6840	0.6160
HH5	0.8105	0.8105	0.8105	0.7570
LH4	0.9280	0.8920	0.9300	0.8115
HL4	0.9280	0.9720	0.9300	0.8940
HH4	0.9809	0.9809	0.9808	0.9740
LH3	0.9810	0.9810	0.8650	0.9808
HL3	0.9810	0.9810	0.8650	0.9500
HH3	0.9808	0.9808	0.6908	0.7890
LH2	0.8650	0.9490	0.3730	0.6903
HL2	0.8650	0.7880	0.3730	0.5020
HH2	0.6903	0.6903	0.1500	0.2320
LH1	0.3730	0.5020	0.0380	0.1498
HL1	0.3730	0.2310	0.0380	0.0590
HH1	0.1498	0.1498	0.0026	0.0080

Table 3.3: Normalized (weights) values for the Subband Weighting Matrices proposals

Subbands	PWM-S1	PWM-S2	PWM-S3	PWM-S4
LH6	2.8428	2.4958	3.4007	2.7694
HL6	2.8428	3.2098	3.4007	3.0868
HH6	3.5233	3.5233	3.9789	3.6969
LH5	4.6045	4.1374	4.8636	3.9753
HL5	4.6045	5.0716	4.8636	4.4666
HH5	5.4087	5.4087	5.5810	5.2705
LH4	6.1926	5.9524	6.2584	5.5814
HL4	6.1926	6.4862	6.2584	6.0516
HH4	6.5455	6.5455	6.5463	6.5077
LH3	6.5463	6.5463	5.8898	6.5463
HL3	6.5463	6.5463	5.8898	6.3709
HH3	6.5447	6.5447	4.9018	5.4529
LH2	5.7722	6.3328	3.1002	4.8900
HL2	5.7722	5.2584	3.1002	3.8166
HH2	4.6062	4.6062	1.8358	2.2772
LH1	2.4891	3.3500	1.2008	1.8087
HL1	2.4891	1.5416	1.2008	1.2908
HH1	1.0000	1.0000	1.0000	1.0000

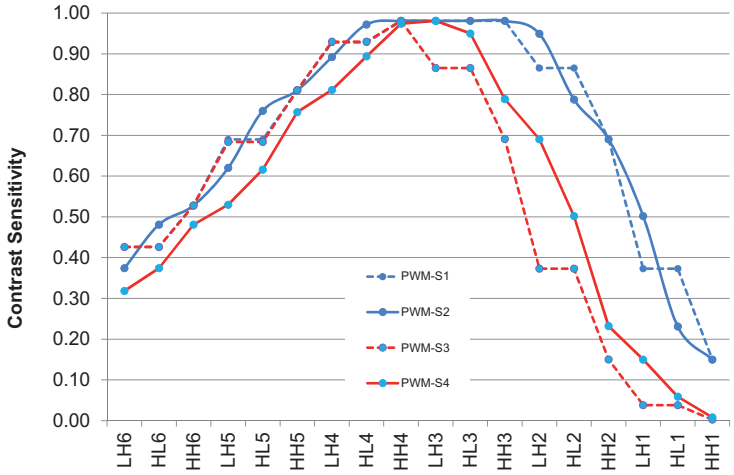


Figure 3.32: Representative subband quantizer values for each proposal

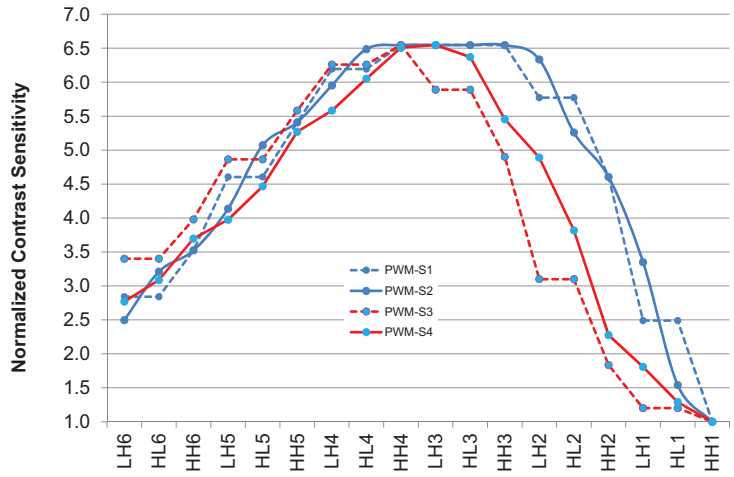


Figure 3.33: Representative normalized weights for each proposal

3.3.1 Weighting matrices performance comparison

Differences among the four proposals are very small for some images, while for others, higher differences are detected. The R/D behavior of the *PWM-S2* and *PWM-S4* proposals is better in the low and mid-compression range than for the remaining proposals, see figures 3.34 and 3.35. The *PWM-S1* and *PWM-S3* proposals perform better in low-compression range. For highly textured images, as Mandril (see Figure 3.35), the *PWM-S2* proposal fails in the low-

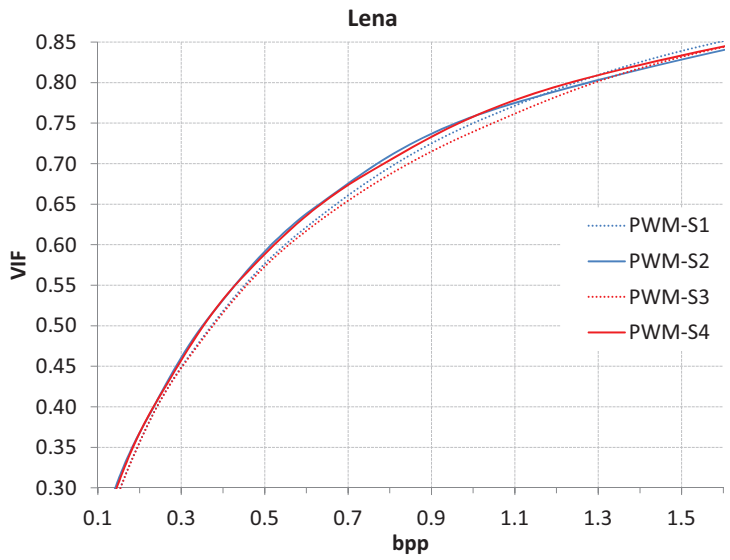


Figure 3.34: Lena: Comparison of the R/D curves for the different proposals.

compression range. Also *PWM-S4* fails with respect to *PWM-S1* and *PWM-S3* in that range, but to a lower extent than *PWM-S2*.

The *PWM-S4* proposal is the one that provides averaged better R/D behavior in our tests, so it is chosen to be compared with the reference proposals, *bLev* (level decomposition) and *bSub* (subband decomposition).

For some of the test images, the R/D curves for the comparison between the *PWM-S4* proposal and the *bSub* and *bLev* proposals are shown in figures 3.36 to 3.39.

By inspecting these figures, we can see that reference proposals cross their R/D curves about the mid-compression rate as stated before. The *PWM-S4* proposal has an overall better R/D performance as it works better in any bit rate range. It provides better VIF values for high and mid-compression rates and almost the same values for low-compression rates than the reference subband proposal, which does not work well for high or mid-compression rates.

As the R/D behavior of the proposals differs depending on the compression range, we have defined three compression ranges in order to obtain the quality gain in each of these ranges as well as for the whole bit rate range. These ranges are:

- High-Compression Range (HCR): from 0 to 0.87 bpp

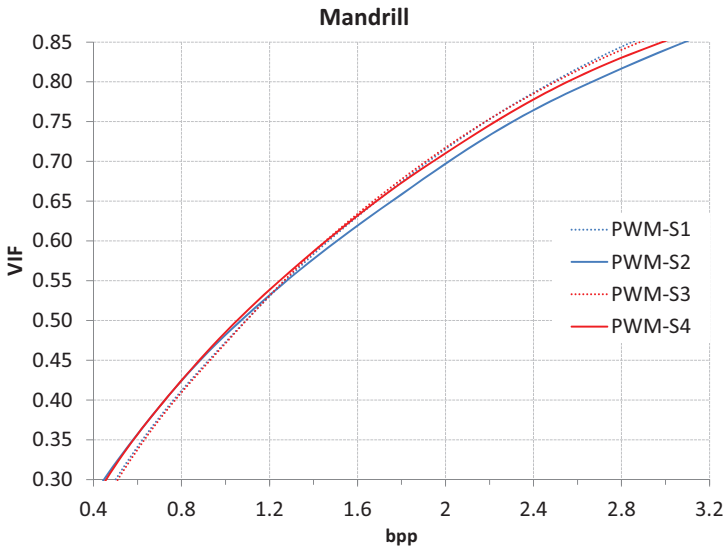


Figure 3.35: Mandrill: Comparison of the R/D curves for the different proposals.

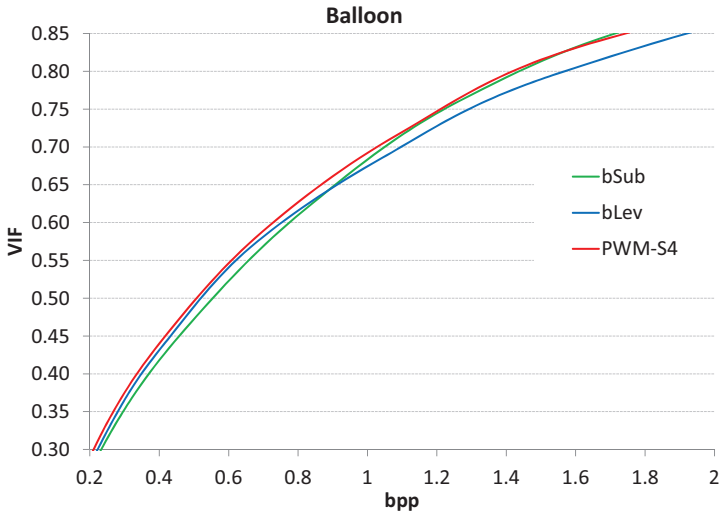


Figure 3.36: Balloon: Comparison of the R/D performance of the PWM-S4 proposal.

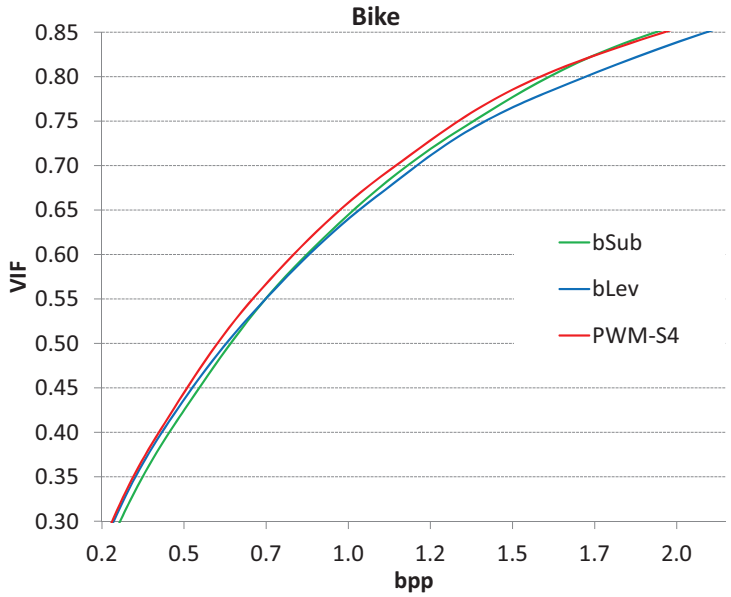


Figure 3.37: Bike: Comparison of the R/D performance of the PWM-S4.

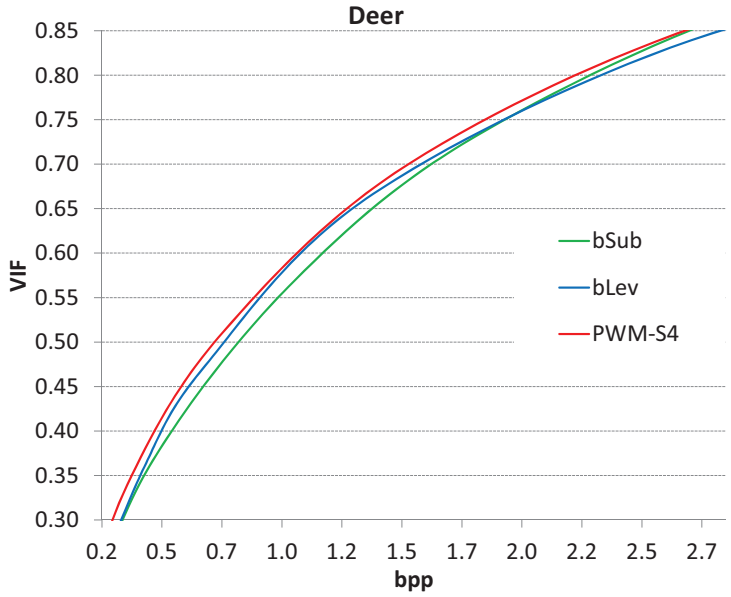


Figure 3.38: Deer: Comparison of the R/D performance of the PWM-S4.

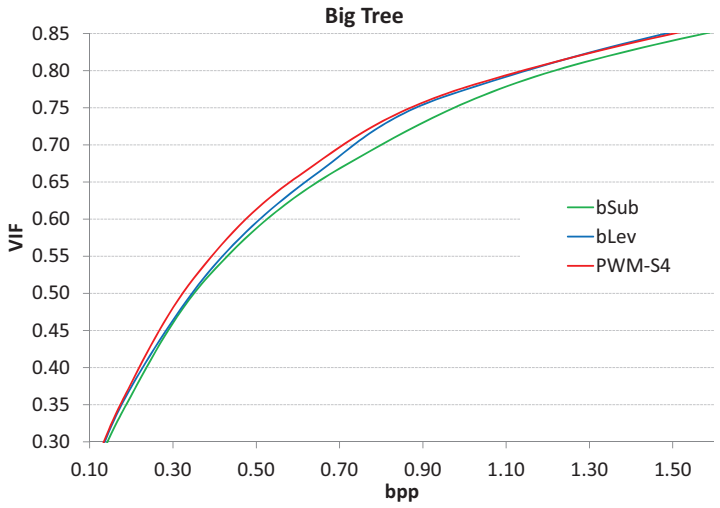


Figure 3.39: Big Tree: Comparison of the R/D performance of the PWM-S4.

- Mid-Compression Range (MCR): from 0.87 to 1.75 bpp
- Low-Compression Range (LCR): from 1.75 to 3.5 bpp

In order to obtain a numerical magnitude of the quality gain or loss in each compression range, we will proceed in the same way as with the Bjontegaard method, see Section 3.2.4, but instead of using PSNR as the distortion metric, we will use the VIF metric.

In order to perform the curve fitting for the typical R/D curve when VIF is used, we propose using 5 real Rate/VIF points evenly distributed along the rate axis. Then, by using Equation 3.29 or Equation 3.30, an accurate curve fitting is obtained. As with the Bjontegaard method, once the parameters p_1, p_2, p_3, and q_1 have been fixed by the curve fitting process, we can get the estimated VIF value for any rate r, and so we can calculate the average gain of one curve over another in a specific rate interval. In the same way, we can calculate the average bit rate saving achieved for a quality interval when the integration limits are set on the VIF axis.

The shape of the R/D curve when the CSF weighting matrix is applied to the encoder, is slightly different than the one obtained when the perceptual weighting is not applied. In particular, the R/D curve with the CSF weighting matrix saturates a bit earlier (on the rate axis) than the other one. The differences seems not to be important when watching both curves, but the curve fitting process reveals that using the proper equation in each case provides better goodness of fit measures. We performed the curve fitting

process with the Matlab curve fitting toolbox, and the goodness of fit parameters that this tool provides are: (for more details see [297]).

- The Sum of Squares due to Error (SSE): Measures the total deviation of the response values from the fit to the response values. It is also called the summed square of residuals and is usually labeled as SSE. A value closer to 0 indicates that the model has a smaller random error component, and that the fit will be more useful for prediction.

- R-square: Measures how successful the fit to explain the variation of the data. Put another way, R-square is the square of the correlation between the response values and the predicted response values. R-square can take on any value between 0 and 1, with a value closer to 1 indicating that a greater proportion of variance is accounted for by the model. For example, an R-square value of 0.8234 means that the fit explains a 82.34% of the total variation in the data around the average.

- Adjusted R-square: It uses the R-square statistic defined above and adjusts it, based on the residual degrees of freedom. The residual degrees of freedom is defined as the number of response values n minus the number of fitted coefficients m estimated from the response values. $v = n - m$ indicates the number of independent pieces of information involving the n data points that are required to calculate the sum of squares. The adjusted R-square statistic can take on any value less than or equal to 1, with a value closer to 1 indicating a better fit.

- Root mean squared error (RMSE): This statistic is also known as the fit standard error and the standard error of the regression. It is an estimate of the standard deviation of the random component in the data. Just as with SSE, a MSE value closer to 0 corresponds to a fit that is more useful for prediction.

In Table 3.4 the mean values for these statistics are shown. The columns *No CSF* and *CSF* indicate if the CSF weighting matrix was applied or not. Not only the data from the fittings performed in this section are computed in Table 3.4, but also those done in further sections, in total 480 fitting processes have been evaluated. So, these goodness of fit statistics confirm that the proposed curves (equations 3.29 and 3.30) are appropriate to estimate the R/D behavior when the distortion metric is VIF, we name these curves VIF R/D curves.

$$VIF(r) = \frac{p_1 \cdot r^2 + p_2 \cdot r + p3}{r + q_1} \tag{3.29}$$

Table 3.4: Goodness of fit for the proposed curve fitting equations.

Goodness of fit	No CSF	CSF
SSE	0.0028	0.0458
R-square	0.9991	0.9996
Adjusted R-square	0.9990	0.9994
RMSE	0.0099	0.0053

$$VIF(r) = \frac{p_1 \cdot r + p_2}{r + q_1} \tag{3.30}$$

A set of comparisons with a large set of images has been made in order to analyze the quality gains or losses for each rate range between the *PWM-S4* proposal and the reference proposals.

First, we will compare the reference proposals among them. In Table 3.5, the subband proposal *bSub* vs. the level proposal *bLev*, are compared. The table shows the image sizes and names (images from 1 to 23 belong to the Kodak Set). The first three columns are the rate ranges and WR is the column for the whole range (from 0 to 3.5 bpp). The results are expressed in percentage of VIF quality gain for the first proposal with respect to the second one. When the percentage is a negative number, then the first proposal has a loss of quality with respect to the second one.

As shown in Table 3.5, *bSub* provides loss of quality for the whole image set in the high compression range, and less than a 1% quality gain on average in the mid-compression range over *bLev* with a maximum of 2.63%. It provides higher quality gains in the low compression-range i.e., for very high bit rates, in average 1.65% gain and up to 4.18% depending on the image.

As both proposals cross their R/D curves in the mid-compression range, as shown in figures 3.36 to 3.39, the gain in the low-compression range is compensated by the loss in the high-compression range, and so taking the whole range in the comparison, only a gain of 0.89% on average is achieved.

The results of comparison of our *PWM-S4* proposal with the subband *bLev* proposal are shown in Table 3.6. In this case, the *PWM-S4* proposal has an average quality gain in all rate ranges, and works better in the high and mid-compression ranges. The lowest average gain is obtained in the low-compression range with 1.68% of gain, and the highest in the mid-compression range with an average of 2.13%. Depending on the image content, the gains go up to a maximum of 4.07% in the high-compression rate, with an average gain of 2.00%. As our proposal gains quality in all ranges, the average quality gain for the whole range is also positive, 1.68% of gain.

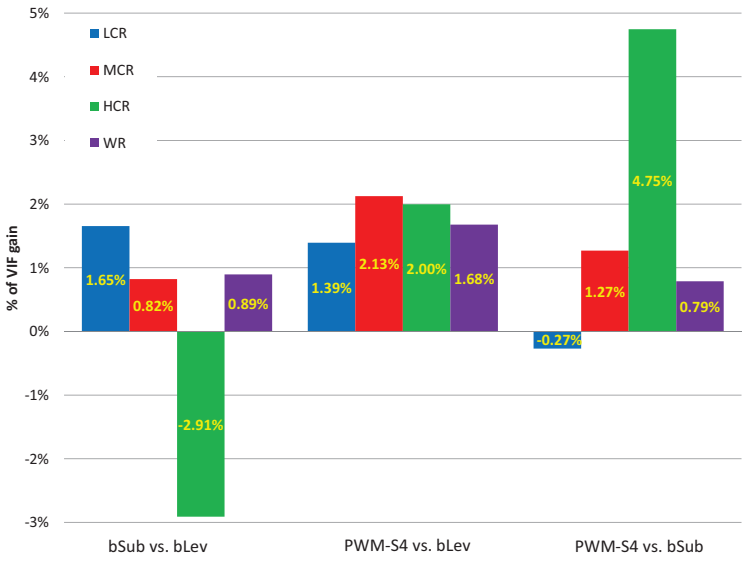


Figure 3.40: Averaged % VIF gains

In Table 3.7, both subbands proposals are compared. In this case, the *PWM-S4* proposal performs better, as expected, than the reference proposal in the high-compression rate with an average gain of 4.75% and in the mid-compression range with an average gain of 1.27%. By contrast, the *PWM-S4* proposal only has a very low loss of quality (0.27%) in the low-compression range. The whole range also obtains a gain of 0.79%.

Figure 3.40 summarizes the average gains for tables 3.5 to 3.7; *bLev* outperforms *bSub* only in the high-compression range, whereas *bSub* is better than *bLev* in the mid- and low-compression ranges.

Our subband proposal *PWM-S4* has an overall better R/D behavior, outperforming both reference proposals in the high- and mid-compression range, and with near the same performance of the best performing reference proposal in the low-compression range.

We have also performed a bit rate saving analysis. We will determine now, how much bit rate can be saved when two images are encoded with the same perceptual quality in terms of the VIF metric. As in the case of the quality gain analysis, we have defined several quality ranges:

- Visually Lossless (VL): $VIF >= 0.83$
- Excellent (E): $0.60 <= VIF <= 0.83$

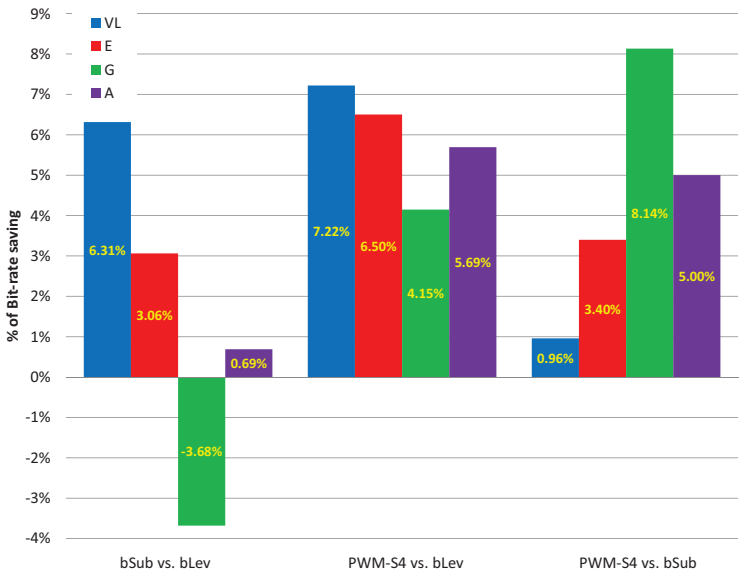


Figure 3.41: Averaged % bit rate savings

- Good (G): 0.30 <= *VIF* < 0.60
- All (A): 0 < *VIF* <= 0.83

Our major interest here is to determine the bit rate savings for at least two approximated quality ranges that we call, *Excellent* and *Good*. Therefore the lower thresholds for these quality ranges have been subjectively selected.

In order to set the lower limit of the *Visually Lossless* range, we have encoded all the images using the corresponding quantization values for the *PQM-S4* proposal, i.e., *PQM-S4* with no further quantization. This produces the at-threshold theoretical image. In Table 3.8, the at-threshold quality level and rate are shown for all images in the test set. The average VIF value for all images encoded at-threshold is used as the lower limit for the VL range.

The quality value for the *Visually Lossless* range is obtained with the comparison between R/D curves at the at-threshold value, i.e., no integration is performed above this threshold as perceptual quality should be the same. As with the Bjontegaard method, the integration of the R/D curves to calculate the bit rate savings in the *Excellent*, *Good*, and *All* ranges, is done over the quality axis.

Figure 3.41 summarizes the comparison of the reference methods and the proposed one.In this figure, the percentage of bit rate saving is the average gain

obtained for all images in the test set for each quality range. The corresponding image data for these figures is available in tables 3.9 to 3.11. Positive values in these tables mean a percentage of bit rate saving, i.e., less bit rate is needed for the same perceptual quality if we use the first method instead of the second one. So, negative values indicate that the first method needs more bit rate to achieve the same perceptual quality.

While comparing in Figure 3.41 the two reference proposals, bSub vs. bLev, is clearly shown that the *bLev* method works better in the *Good* range, while the *bSub* method is better in the *Excellent* range and at the *Visually Lossless* threshold. The bit rate savings in the *All* range is practically canceled because the gain in one range is compensated with the loss in the other. This is shown in Figure 3.36, where both R/D curves cross approximately between the *Excellent*, and *Good* ranges. This is repeated in most of the images in the test stet.

Figure 3.41 also shows that when using of the proposed *PWM-S4* method in the S-LTW encoder, bit rate savings are obtained in all the quality ranges with respect to both reference methods.

In the comparisons, *PWM-S4* obtains higher savings with respect to *bLev* than with bSub, except for the *Good* range, because as said before, *bSub* works better in the *Excellent* range than in the *Good* range. However, *PWM-S4* obtains averaged bit rate savings of 4.15% in the *Good* range. At *Visually Lossless* threshold it saves in average *7.22%*, in the *Excellent* range *6.50%* and in all the range *5.69%*. Theses are averaged values for all images, but for some images higher gains are obtained, so for example for image number 13 of the Kodak set and for the Bike image, 11.50% and 10.49% of bit rate saving, respectively, is obtained at *Visually Lossless* threshold, and 10.08% in the *Excellent* range for the Bike image. In the *Good* range the maximum saving is obtained for the Deer image with 7.64%.

As the bSub reference when used with the S-LTW, works much better than bLev, the bit rate savings obtained in the comparison between *PWM-S4* and bSub at *Visually Lossless* threshold are not as high as when compared with *bLev*. But, as *PWM-S4* proposal works good in the whole the quality range, not only in the *Excellent* range or in the *Good* range, it get also bit rate savings when compared with *bSub*. So, at *Visually Lossless* threshold only an averaged bit rate saving of 0.96% is obtained, while for the *Excellent* range the savings are 3.40%, and much higher for the *Good* range with 8.14%. This produces averaged bit rate savings of 5.00% in the *All* range. As before, these values were averaged for all images, and higher values are obtained depending on the image. So, for example, at *Visually Lossless* threshold, 8.73% bit rate saving can be obtained for the Big Tree image, and the maximum for the *Excellent*

and *Good* ranges are obtained for Zelda with 9.37%, and for Deer with 13.41% respectively.

As shown, the proposed PWM-S4 perceptual weighting matrix with subband granularity, provides better results than the reference weighting matrices regardless of the quality range being used.

Table 3.5: bSub vs. bLev % of quality gain.

bSub vs. bLev (% VIF Gain)					
Image Size	Image	HCR	MCR	LCR	WR
7168x5376	Big building	-1.65	0.00	0.44	0.00
6016x4480	Big tree	-1.97	-1.44	-0.50	-0.97
3968x2560	Deer	-3.71	-0.66	0.16	-0.55
2048x2560	Bike	-3.82	Âº	4.18	2.83
	Woman	-0.79	1.39	1.03	0.87
	Cafe	-1.94	0.94	0.64	0.32
768x512	1	-2.41	0.49	0.79	0.26
	2	-5.17	0.01	2.32	0.94
	3	-3.40	2.23	3.31	2.26
	4	-1.80	0.18	0.71	0.19
	5	-4.28	1.31	3.83	2.35
	6	-1.12	1.61	1.05	0.87
	7	-1.65	1.23	1.14	0.76
	8	-2.29	1.86	2.38	1.66
	9	-2.09	1.05	0.99	0.56
	10	-7.98	-0.41	3.90	1.75
	11	-3.78	0.63	2.00	0.97
	12	-2.85	1.06	1.07	0.49
	13	-2.28	1.97	1.97	1.41
	14	-3.23	0.01	0.72	-0.03
	15	-4.17	0.85	2.44	1.27
	16	-3.07	1.26	1.87	1.07
	17	-0.43	2.42	1.06	1.18
	18	-2.27	2.63	2.90	2.18
	19	-2.33	1.27	1.75	1.10
	20	-1.53	0.87	0.35	0.17
	21	-2.10	1.28	1.80	1.15
	22	-0.96	2.47	2.42	2.01
	23	-3.71	1.31	3.43	2.14
512x512	Lena	-2.15	-0.60	-0.08	-0.53
	Zelda	-2.07	-0.58	-0.18	-0.61
	Barbara	-5.75	-1.06	0.71	-0.56
	Balloon	-2.54	1.79	2.16	1.43
	Boat	-1.57	1.29	1.44	0.97
	Mandrill	-9.05	-0.73	3.70	1.40
Mean		**-2.91**	**0.82**	**1.65**	**0.89**

Table 3.6: PWM-S4 vs. bLev. % of quality gain.

PWM-S4 vs. bLev (% VIF Gain)					
Image Size	Image	HCR	MCR	LCR	WR
7168x5376	Big building	1.54	1.09	0.55	0.84
6016x4480	Big tree	1.63	0.50	0.18	0.50
3968x2560	Deer	4.07	1.77	0.71	1.44
2048x2560	Bike	2.80	3.84	3.15	3.27
	Woman	3.50	2.49	1.12	1.80
	Cafe	1.82	1.93	0.71	1.20
768x512	1	2.29	1.65	0.79	1.23
	2	1.95	2.41	2.02	2.11
	3	2.11	3.65	2.86	2.96
	4	1.64	1.23	0.65	0.95
	5	1.65	3.06	2.94	2.82
	6	2.19	2.20	0.98	1.47
	7	1.92	1.99	1.02	1.40
	8	2.59	2.88	1.88	2.22
	9	2.10	2.17	1.07	1.50
	10	1.04	2.90	3.37	3.01
	11	2.58	2.57	1.80	2.09
	12	1.44	1.60	0.75	1.07
	13	2.79	3.31	1.91	2.39
	14	1.57	1.28	0.69	0.98
	15	2.27	2.54	1.89	2.10
	16	2.14	2.40	1.51	1.82
	17	2.44	2.33	0.70	1.39
	18	2.64	3.45	2.32	2.64
	19	2.66	2.22	1.28	1.70
	20	1.81	1.68	0.41	0.97
	21	2.17	2.36	1.51	1.81
	22	3.20	2.95	1.70	2.21
	23	1.81	2.94	2.71	2.66
512x512	Lena	1.53	0.53	0.17	0.48
	Zelda	1.38	0.94	0.30	0.65
	Barbara	-1.89	-1.50	-0.88	-1.17
	Balloon	1.96	2.47	1.60	1.87
	Boat	2.44	2.11	1.19	1.61
	Mandrill	0.12	2.49	3.20	2.70
Mean		**2.00**	**2.13**	**1.39**	**1.68**

Table 3.7: PWM-S4 vs. bSub. % of quality gain.

PWM-S4 vs. bSub (% VIF Gain)					
Image Size	Image	HCR	MCR	LCR	WR
7168x5376	Big Building	3.14	1.10	0.11	0.84
6016x4480	Big tree	3.53	1.91	0.67	1.46
3968x2560	Deer	7.50	2.42	0.55	1.98
2048x2560	Bike	6.37	1.56	-1.07	0.46
	Woman	4.25	1.12	0.09	0.95
	Cafe	3.69	0.99	0.06	0.88
768x512	1	4.59	1.16	0.00	0.97
	2	6.77	2.41	-0.31	1.18
	3	5.33	1.45	-0.48	0.71
	4	3.37	1.05	-0.06	0.76
	5	5.69	1.77	-0.92	0.48
	6	3.27	0.60	-0.06	0.61
	7	3.51	0.77	-0.13	0.65
	8	4.78	1.03	-0.51	0.57
	9	4.11	1.13	0.08	0.94
	10	8.35	3.30	-0.56	1.29
	11	6.12	1.95	-0.20	1.13
	12	4.17	0.54	-0.32	0.58
	13	4.96	1.38	-0.06	0.99
	14	4.65	1.27	-0.03	1.01
	15	6.18	1.71	-0.56	0.83
	16	5.05	1.15	-0.37	0.76
	17	2.85	-0.10	-0.36	0.21
	18	4.80	0.84	-0.60	0.48
	19	4.88	0.96	-0.48	0.61
	20	3.28	0.82	0.07	0.81
	21	4.18	1.09	-0.30	0.67
	22	4.12	0.49	-0.74	0.20
	23	5.32	1.65	-0.75	0.53
512x512	Lena	3.60	1.12	0.25	1.01
	Zelda	3.37	1.51	0.48	1.25
	Barbara	3.65	-0.44	-1.59	-0.60
	Balloon	4.38	0.68	-0.58	0.44
	Boat	3.95	0.83	-0.25	0.64
	Mandrill	8.40	3.20	-0.52	1.32
Mean		**4.75**	**1.27**	**-0.27**	**0.79**

Table 3.8: At-threshold quality and rate for images encoded with S-LTW and PQM-S4 matrix, with no further quantization.

Image Size	Image	Rate	VIF
7168x5376	Big building	1.42	0.87
6016x4480	Big tree	1.33	0.83
3968x2560	Deer	1.61	0.72
2048x2560	Bike	1.88	0.85
	Woman	1.81	0.80
	Cafe	2.48	0.85
768x512	01	2.55	0.83
	02	1.62	0.80
	03	1.30	0.84
	04	1.61	0.82
	05	2.43	0.86
	06	2.05	0.83
	07	1.44	0.88
	08	2.63	0.84
	09	1.44	0.80
	10	1.54	0.82
	11	1.95	0.83
	12	1.52	0.82
	13	2.86	0.82
	14	2.22	0.84
	15	1.48	0.82
	16	1.69	0.83
	17	1.58	0.85
	18	2.20	0.83
	19	1.84	0.84
	20	1.29	0.84
	21	1.89	0.82
	22	1.89	0.83
	23	1.14	0.82
512x512	Lena	1.54	0.84
	Zelda	1.19	0.83
	Barbara	1.90	0.83
	Balloon	1.82	0.85
	Boat	1.70	0.84
	Mandrill	2.76	0.82
	Man VIF at-threshold		**0.83**

Table 3.9: bSub vs. bLev. % of bit rate savings.

bSub vs. bLev (% Rate)					
Image Size	Image	VL	E	G	A
7168x5376	Big building	-0.11	-1.31	-3.55	-2.07
6016x4480	Big tree	-6.09	-5.30	-3.62	-4.78
3968x2560	Deer	0.81	-0.36	-5.09	-1.87
2048x2560	Bike	14.35	10.15	-1.10	6.05
	Woman	5.99	4.13	-1.05	2.40
	Cafe	4.30	0.86	-5.28	-1.18
768x512	1	3.62	0.86	-4.55	-0.94
	2	6.56	2.73	-4.33	0.08
	3	11.99	7.30	-2.39	3.72
	4	0.36	-0.88	-3.57	-1.81
	5	12.08	7.55	-1.86	4.02
	6	7.72	4.89	-2.67	2.47
	7	6.36	3.06	-3.48	0.89
	8	9.46	5.64	-2.22	2.83
	9	5.63	2.47	-4.04	0.28
	10	12.52	7.78	-3.46	3.47
	11	6.77	3.19	-3.82	0.67
	12	6.17	2.04	-6.09	-0.66
	13	9.39	5.23	-3.20	2.26
	14	1.17	-1.55	-6.16	-3.13
	15	8.36	4.25	-3.78	1.33
	16	7.83	3.61	-4.50	0.78
	17	10.03	5.50	-2.79	2.67
	18	12.71	8.29	-1.89	4.70
	19	7.28	3.97	-2.89	1.58
	20	4.74	0.80	-6.11	-1.32
	21	6.95	3.46	-3.00	1.20
	22	10.51	7.34	-0.14	4.69
	23	10.55	6.31	-1.94	3.22
512x512	Lena	-2.11	-2.69	-4.27	-3.19
	Zelda	-2.90	-4.00	-5.91	-4.55
	Barbara	0.27	-2.68	-7.98	-4.57
	Balloon	8.92	4.27	-4.23	1.29
	Boat	6.55	3.22	-2.95	1.13
	Mandrill	12.25	7.06	-4.87	2.52
Mean		**6.31**	**3.06**	**-3.68**	**0.69**

Table 3.10: PWM-S4 vs. bLev. % of bit rate savings.

PWM-S4 vs. bLev (% Rate)					
Image Size	Image	VL	E	G	A
7168x5376	Big building	4.52	4.10	3.18	3.79
6016x4480	Big tree	2.11	3.07	4.40	3.48
3968x2560	Deer	2.35	3.90	7.64	5.02
2048x2560	Bike	10.49	10.08	6.23	8.74
	Woman	7.44	7.93	7.05	7.65
	Cafe	8.51	6.76	3.15	5.58
768x512	1	5.68	5.62	4.68	5.32
	2	6.85	5.81	3.63	5.02
	3	11.75	9.74	4.52	7.87
	4	4.83	4.49	3.45	4.13
	5	9.32	8.01	4.23	6.64
	6	8.33	7.87	4.47	6.82
	7	7.83	6.73	3.68	5.74
	8	8.85	7.89	4.96	6.87
	9	8.09	7.06	3.99	6.05
	10	9.39	8.32	3.95	6.71
	11	7.37	6.68	4.74	6.01
	12	6.21	5.41	2.77	4.56
	13	11.50	9.78	5.24	8.22
	14	4.85	4.25	3.02	3.84
	15	7.44	6.61	4.29	5.80
	16	8.18	7.11	4.12	6.10
	17	9.82	8.16	4.43	6.93
	18	11.29	10.00	5.35	8.42
	19	6.55	6.35	5.01	5.90
	20	9.09	7.40	3.24	6.15
	21	7.81	6.73	4.21	5.87
	22	8.19	7.95	5.92	7.26
	23	8.89	7.45	4.05	6.21
512x512	Lena	1.80	2.97	4.25	3.36
	Zelda	5.38	5.00	3.36	4.52
	Barbara	-4.00	-3.82	-3.33	-3.65
	Balloon	8.82	7.17	3.64	5.97
	Boat	7.66	6.91	4.87	6.24
	Mandrill	9.52	8.02	2.80	6.11
Mean		**7.22**	**6.50**	**4.15**	**5.69**

Table 3.11: PWM-S4 vs. bSub. % of bit rate savings.

PWM-S4 vs. bSub (% Rate)					
Image Size	Image	VL	E	G	A
7168x5376	Big building	4.64	5.49	6.98	5.99
6016x4480	Big tree	8.73	8.84	8.32	8.68
3968x2560	Deer	1.54	4.27	13.41	7.02
2048x2560	Bike	-3.37	-0.07	7.40	2.53
	Woman	1.37	3.66	8.19	5.12
	Cafe	4.04	5.84	8.89	6.84
768x512	1	1.98	4.72	9.67	6.32
	2	0.27	3.00	8.32	4.93
	3	-0.21	2.27	7.08	4.00
	4	4.46	5.42	7.28	6.06
	5	-2.46	0.42	6.20	2.52
	6	0.56	2.85	7.34	4.24
	7	1.38	3.56	7.42	4.81
	8	-0.55	2.13	7.34	3.93
	9	2.32	4.48	8.36	5.76
	10	-2.78	0.50	7.67	3.13
	11	0.56	3.39	8.90	5.30
	12	0.04	3.30	9.43	5.26
	13	1.93	4.33	8.71	5.84
	14	3.64	5.89	9.78	7.19
	15	-0.85	2.27	8.38	4.41
	16	0.33	3.38	9.02	5.28
	17	-0.19	2.53	7.43	4.15
	18	-1.26	1.58	7.39	3.56
	19	-0.68	2.29	8.13	4.25
	20	4.16	6.54	9.95	7.57
	21	0.81	3.16	7.43	4.61
	22	-2.10	0.57	6.07	2.45
	23	-1.50	1.07	6.11	2.91
512x512	Lena	3.99	5.81	8.90	6.76
	Zelda	8.53	9.37	9.85	9.51
	Barbara	-4.26	-1.17	5.05	0.97
	Balloon	-0.09	2.78	8.21	4.62
	Boat	1.04	3.57	8.06	5.05
	Mandrill	-2.43	0.90	8.06	3.51
Mean		**0.96**	**3.40**	**8.14**	**5.00**

3.4 Perceptually Enhanced Tree Wavelet codec (PETW)

The LTW and the S-LTW encoders employ a quantization mechanism based on two parameters [226], one finer (Q) and another coarser (*rplanes*). Thus, the quantized image is the result of jointly applying two quantization methods. The first method performs a scalar quantization with a step-size of $2Q$, and the second one consists in removing the *rplanes* least significant bits of all coefficients, being a simple bit-plane quantization process.

In this section, we will introduce a new encoder proposal called Perceptually Enhanced Tree Wavelet (PETW) based also on the S-LTW encoder that besides having the perceptual weighting stage (by the use of the PWM), it has a new quantization strategy based on a Uniform Variable Dead Zone Quantizer (UVDZQ), so it can reduce in one the parameters needed by the encoder to control the quantization stage. The S-LTW has, by contrast, two parameters to control the quantization stage, *rplanes* that controls the coarse quantization and Q that controls the uniform quantizer.

Setting in the UVDZQ the equivalent dead zone size that the S-LTW uses, allow us to use only the step-size parameter Q to control the amount of quantization, providing the same results as when no perceptual enhancement is applied. The quantizer change also enables us to obtain encoded images with higher rates than with the S-LTW encoder, as we do not have the restriction imposed by the coarse quantizer that is always applied with a minimum value of $rplanes = 2$. Reaching higher rate ranges is appropriate, as told in 3.3, for working in the sub-threshold area or visually lossless, where distortions are supposed not to be detected in static images.

The motivation for changing the S-LTW quantization stage is based on the work of [298], where authors made several performance comparisons between a Uniform Scalar Quantizer (USQ), a Uniform Scalar Dead Zone Quantizer (USDZQ), and a Universal Trellis Coded Quantizer (UTCQ), using the same step size, and applied to DWT and DCT transformed coefficients. Their performance comparisons show that the UTCQ can quantize data more precisely and provide better PSNR results than the other two quantizers when using the same step size. But, when they are combined with zero or higher order entropy coders, the dead zone quantizer (the USDZQ) is the best instead. In these comparisons, the authors show that if the dead zone is designed carefully, the USDZQ can effectively reduce significantly the output hits of the entropy coder, and although it reduces quantization precision by discarding some data around zero, the obtained rate reduction is worthwhile. Moreover, the USDZQ is only a USQ with a dead zone, and its computational complexity

is lower than the UTCQ.

Our studies are oriented, by contrast, to optimize the R/D behavior in terms of the VIF QAM, and hence to determine which is the influence of the dead zone size over the perceptual quality, and not over the PSNR as in previous studies.

The variable deadzone schema that we will use in the PETW encoder is the one proposed in the JPEG2000 encoder [299, 274]. So the first step to change in the S-LTW to obtain the PETW encoder is to include the PWM in it. This has been widely explained before.

In the following subsections, we will briefly review the S-LTW 2-stage quantizer in contrast with the UVDZQ, and we will also overview how the quantizer change has been made. We will also prove that the new PETW encoder has the same PSNR performance than the S-LTW when no perceptual enhancements are applied, because both quantization strategies are equivalent.

Then, we will also test the performance of the new PETW encoder with video sequences encoded in intra mode, and make some comparison with standard video encoders also working in intra mode.

3.4.1 PETW quantizer

As we will see later, the two quantization stages of the S-LTW act jointly as a Uniform Dead Zone Quantizer (UDZQ), and therefore the use of both quantization processes may seem a bit strange, but it reveals more natural when the LTW is studied in depth, as some optimizations can be included as result [226]. The coarser quantization is useful to shorten the number of bits needed to represent a coefficient, and to concentrate the symbol probability. In addition, it allows the introduction of quantization in architectures that only support integer arithmetic. Finally, with this type of quantization, some values are never employed by significant coefficients (in particular those $\left|C_{i,j}\right| < 2^{rplanes}$), and this range is used to represent specific marks and control symbols (such as LOWER and ISOLATED_LOWER), allowing in-place symbol computation (which avoids the introduction of extra memory to store those symbols). In the bit-plane quantization, the available step-sizes are always powers of two, and thus its granularity is very low. Therefore, a fine control of the image compression is not possible with only this quantization parameter. In order to perform finer rate control, a scalar quantization stage is required.

Changing the encoder in the S-LTW encoder is not a trivial substitution because the whole encoder is designed and based on the existence of these

two quantization stages. The design of the quantization strategy of the S-LTW encoder has a big influence on all parts of the source code and the design of other parts in the coding chain. For example, as the coefficients have been truncated in their two less significant bits, in order to reconstruct the coefficient value with half the error, a value of $2^{rplanes-1}$ is added in the dequantization process. As the *rplanes=2* truncation, sets all coefficient values above or equal to 4, then the coarse quantized coefficient values will never be in the range from -3 to 3, and this fact is guiding the source code of several parts in the encoder.

All these algorithms must be changed if we want to maintain these truncated bits as part of the coefficient value, and then we could operate in lower compression rate ranges. The UVDZQ allows these lower significant bits to still belong to the coefficient value as it does not impose any bit truncation operation. But then these special symbols that drive the encoder must be set in another part of the encoder, and so, we are forced to add new symbols to the encoder symbol map. This produces a bit memory overhead and slightly reduces the performance of the arithmetic encoder, but this it is highly compensate by the quality gain introduced by the PWM, and the optimization of the dead zone. So, the new PETW encoder must change many important parts of the original source code.

Details of the quantizer substitution process and other changes made in the source code are omitted here for brevity, for more detail about the internals of the S-LTW and the LTW please refer to [226]. At the end, from a quantization point of view, the most important issue is to determine the parameters of the UVDZQ that produce the same results as the two stages of the S-LTW together, when the step size for both strategies is the same. In [226], Oliver exposes how the two stage quantization is performed in the LTW. We will briefly review this formulation and compare it with that of the UVDZQ in order to determine when and why both strategies are equivalent. The formulation for the S-LTW quantization is reproduced in equations 3.31 to 3.37.

Let us call c the initial wavelet coefficient, c_Q the quantized coefficient, and c_R a coefficient recovered on the decoder side. Then, the use of the two quantization stages can be mathematically expressed with equations 3.31 to 3.33, for the forward quantization, and equations 3.34 to 3.36 for the reverse quantization or dequantization.

$$if \;\; c > 0 \quad c_Q = \left\lfloor \frac{\left(\left\lfloor \frac{c}{2Q} + 0.5 \right\rfloor + K \right)}{2^{rplanes}} \right\rfloor \tag{3.31}$$

$$if \;\; c < 0 \quad c_Q = \left\lfloor \frac{\left(\left\lceil \frac{c}{2Q} - 0.5 \right\rceil + K \right)}{2^{rplanes}} \right\rfloor \tag{3.32}$$

$$if \;\; c = 0 \quad c_Q = 0 \tag{3.33}$$

Note that an integer constant K can be used to adjust the bit plane quantization (by taking some values out of the dead zone, i.e., narrowing it), which may be useful in some cases. Experimental tests by the LTW authors have revealed that $K = 1$ is a good value, increasing the PSNR R/D performance for most source images.

$$if \;\; c_Q > 0 \quad c_R = \left(2\left((2c_Q + 1)\,2^{rplanes} - K\right) - 1\right) Q \tag{3.34}$$

$$if \;\; c_Q < 0 \quad c_R = \left(2\left((2c_Q - 1)\,2^{rplanes} + K\right) + 1\right) Q \tag{3.35}$$

$$if \;\; c_Q = 0 \quad c_R = 0 \tag{3.36}$$

In both dequantization processes, i.e., in the standard scalar dequantization and in the dequantization from the bit-plane removing, the c_R value is adjusted to the midpoint within the recovering interval, reducing in this way the quantization error.

The equations for the dequantization process may be clearer if we observe both dequantization processes separately. First, we have to recover the initial number of bits of the scalar quantized coefficient; thus if $c_Q > 0$ in Equation 3.37, the value c'_R is the temporal value obtained from the coarse dequantization, and with Equation 3.38, we finally obtain the value of the recovered coefficient c_R.

$$c'_R = c_Q 2^{rplanes} + 2^{rplanes-1} = (2c_Q + 1)\,2^{rplanes-1} \tag{3.37}$$

$$c_R = \left(2\left(c'_R - K\right) - 1\right) Q \tag{3.38}$$

As stated before, the quantization of the S-LTW that is jointly produced by the coarser and finer quantizers produces a dead zone and therefore it could be substituted with UDZQ. However, such quantizer has a dead zone size of 2Δ, being Δ the quantization step size. But in the S-LTW, this dead zone size has been changed by the use of parameter K in equations 3.31,3.32, 3.34, and 3.35.

We can instead use a UVDZQ so we can control the size of the dead zone. So, in order to change the S-LTW quantizers with a UVDZQ, the first step is to find the correspondence with UVDZQ. We will reformulate the S-LTW quantization so that it is easier to get this correspondence.

The overall step size Δ applied to S-LTW can be viewed as the multiplication of two deltas, Δ_1 corresponding to the finer quantizer and Δ_2 corresponding to the coarser one, as shown in Equation 3.39.

$$\begin{aligned} \Delta &= \Delta_1 \cdot \Delta_2 \\ \Delta_1 &= 2Q \\ \Delta_2 &= 2^{rplanes} \end{aligned} \tag{3.39}$$

In order to replace these two quantizers with a UVDZQ, we must know the relationship between the dead zone size, and the overall Δ applied in S-LTW, i.e., to obtain the equivalent dead zone size in the UVDZQ that is used in S-LTW.

To search for this relationship we can use equations 3.40, where U stands for the upper bound (maximum positive value) of the dead zone in S-LTW. The Dead Zone (DZ) size is therefore $DZ = 2U$, and the relationship between the dead zone size and the overall Δ is determined with the τ constant.

The value τ depends on the values of Δ_1 and Δ_2, see Equation 3.39. When the value of Δ_2 is fixed, i.e., the $rplanes$ parameter is fixed, then it is easy to see that the τ constant keeps the same value for increasing values of Δ_1. For example if we fix $rplanes = 2$ (which is the minimum $rplanes$ allowed in S-LTW) then $\tau = 1.25$ independently of the Δ_1 step size of the finer quantizer.

So, because the K parameter is fixed in S-LTW to $K = 1$, we can obtain the value of τ by fixing the value of Δ_2 (fixing the value of the $rplanes$).

For example, for $rplanes = 2$, if we set $\Delta_1 = 1$, i.e., no finer quantization ($Q = 0.5$, see Equation 3.2), then $\Delta = \Delta_1 \cdot \Delta_2 = 1 \cdot 4$, then $U = 2.5$ and hence the dead zone size is $DZ = 5$. Finally $\tau = DZ/\Delta = 5/4 = 1.25$.

It is easy to see that this relationship holds for increasing values of Δ_1. In Table 3.12, the values of τ for different $rplanes$ are shown. This way we can determine the dead zone size whatever $rplanes$ is used in S-LTW, but as our objective is to suppress the $rplanes$ based quantizer, we will use the value $\tau = 1.25$, i.e., we will fix $rplanes = 2$ as in the original LTW version.

Table 3.12: Relationship between the dead zone size and the overall step size Δ, depending on the *rplanes* value.

rplanes	DZ size
2	$DZ = 1.25\Delta$
3	$DZ = 1.63\Delta$
4	$DZ = 1.81\Delta$
5	$DZ = 1.91\Delta$
6	$DZ = 1.95\Delta$

$$\begin{aligned} U &= \Delta_1 \cdot (\Delta_2 - (K + 0.5)) \\ DZ &= 2U \\ DZ &= \tau\Delta \end{aligned} \tag{3.40}$$

We reformulate the S-LTW quantization as a UDZQ so we can see the relationship with the dead zone. We also use here a ρ constant that enables us to round or truncate the coefficients in the S-LTW quantizer, see Equation 3.41. We have separated the formulation of the forward quantization for the finer and coarser quantizers, with equations 3.42 and 3.43, respectively, where $>> rplanes$ is a bit displacement of *rplanes* bits to the right and $\lfloor . \rfloor$ is the truncation operation.

$$\rho = \begin{Bmatrix} 0 & for\ truncating \\ 0.5 & for\ rounding \end{Bmatrix} \tag{3.41}$$

$$c_1 = \begin{cases} sign(c) \left\lfloor \frac{|c|}{\Delta_1} + K + \rho \right\rfloor & if \left| \frac{c}{\Delta_1} \right| \geq U \\ 0 & if \left| \frac{c}{\Delta_1} \right| < U \end{cases} \tag{3.42}$$

$$c_Q = c_1 >> rplanes \tag{3.43}$$

The inverse quantization steps in S-LTW can also be expressed with separate expressions. The value δ sets the recovering point inside the step size, so a value of $\delta = 0.5$ sets the recovering point in the centroid of the interval. Equation 3.44 provides the intermediate c_1 value after the inverse coarse quantization, where $<< n$ stands for a left bit displacement operator of n bits and | is the bitwise OR operator, and Equation 3.45 gives the final recovered coefficient after the finer inverse quantization.

$$c_1 = sign(c_Q)\left[(c_Q << rplanes) \,|\, (1 << (rplanes - 1))\right] \tag{3.44}$$

$$c_R = (c_1 - (K + \delta))\,\Delta_1 = c_1\Delta_1 - \Delta_1\,(K + \delta) \tag{3.45}$$

Once we have determined the DZ size in relation with the overall step size Δ of S-LTW, we must set the correct parameters in the UVDZQ formulation, so that its dead zone size is also $DZ = 1.25\Delta$. Then, we can check if the results obtained with the PETW encoder without perceptual enhancements but with the UVDZQ are the same as those obtained with S-LTW. Finally, we will proceed with the performance analysis.

As with the case of the S-LTW quantization, the ρ parameter determines if we will finally use a truncation operation or a rounding one in the quantizer, see Equation 3.41, and also the $0 \leq \delta < 1$ parameter sets the recovering point inside the quantization interval. Equation 3.46 is the forward quantizer expression for a UVDZQ that sets the value c_Q of the quantized coefficient [274]. The parameter ξ, so that $\xi < 1$, determines the size of the dead zone in such a quantizer. Depending on the value of this parameter, the dead zone size is set as follows:

- $\xi < 0$ increases the dead zone size above the size of 2Δ
- $\xi = 0$ produces a dead zone with double the size as the quantization step, i.e., 2Δ and then the upper bound of the positive part of the dead zone is Δ
- $0 < \xi < 1$ reduces the dead zone so that its size is lower than 2Δ. A typical value is $\xi = 0.500$, which produces a dead zone size of Δ

$$c_Q = \begin{cases} sign(c)\left\lfloor \frac{|c|+\xi\Delta}{\Delta} + \rho \right\rfloor & if\ \frac{|c|}{\Delta} + \xi + \rho\ > 0 \\ 0 & Otherwise \end{cases} \tag{3.46}$$

$$c_R = sign(c)\left(\left|c_Q\right| - \xi + \delta - \rho\right)\Delta \tag{3.47}$$

So, if for S-LTW we have a τ constant of 1.25, i.e., a dead zone size of 1.25Δ, we must use a ξ value so that $0 < \xi < 1$. To use a UVDZQ that equals the behavior of both quantizers of the S-LTW acting together we must fix $\xi = 0.375$ and $\rho = 0$. With Equation 3.47, we can finally obtain the reconstructed coefficient c_R.

In figures 3.42 and 3.43, we can see that the PSNR R/D behavior of the equivalent PETW, with a dead zone size of $DZ = 1.25\Delta$ obtained with a $\xi = 0.375$, is almost the same as the one obtained with the original joint quantization of S-LTW.

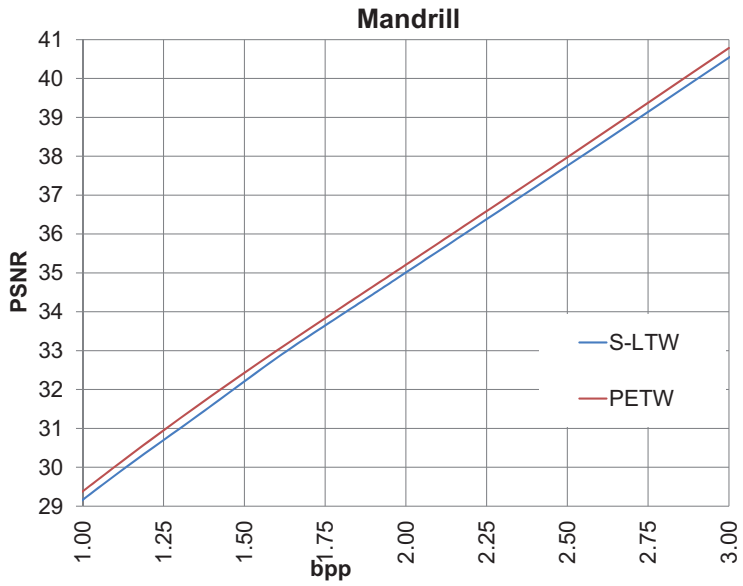


Figure 3.42: Equivalence of the R/D behavior between S-LTW joint quantization and the PETW dead zone quantization for Mandrill.

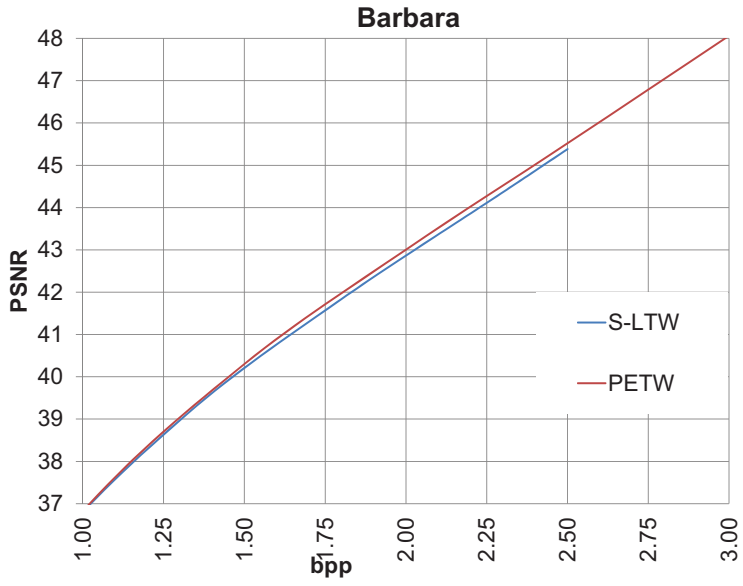


Figure 3.43: Equivalence of the R/D behavior between S-LTW joint quantization and the PETW dead zone quantization for Barbara.

For some images with high frequency content, as those shown in Figure 3.42, the obtained PSNR is slightly better with the new quantization schema of PETW. This is because in S-LTW, the rate control is enabled and the *rplanes* parameter changes depending on the desired rate. This is an expected result as stated by the S-LTW authors because fixing the $rplanes = 2$ and then increasing the finer quantizer up to the desired rate produces slightly better results.

In Figure 3.43, we can also see that when using the UVDZQ in the PETW encoder we can obtain values with lower compression, i.e., in the sub-threshold area, than when using the S-LTW encoder.

3.4.2 Performance results for video sequences encoded in intra mode

Currently, most popular video compression technologies operate in both intra and inter coding modes. Intra mode compression operates on a frame-by-frame basis, while inter mode works with a Group Of Pictures (GOP) at a time.

Inter mode compression is able to achieve higher coding efficiency than intra mode schemes when picture content of adjacent frames is quite similar. However, under certain conditions, such as fast camera zooms and pans, high intensity motion (sports, animation, etc.), still camera flash lights, and strobe lights, as well as other short duration production effects, the correlation of adjacent frames is severely reduced and results in a visibly reduced picture quality, or at worst, blocking artifacts.

Most television content productions require recordings in HD to maintain high picture quality even though the usual final transmission is in SD format. In video content production stages, digital video processing applications require fast frame random access to perform an undefined number of real-time *decompressing-editing-compressing* interactive operations without a significant loss of original video content quality.

Intra-frame coding is desirable as well in many other applications, like video archiving, high-quality high-resolution medical and satellite video sequences, applications requiring simple and fast real-time encoding, like video-conference and video surveillance systems [300], and Digital Video Recording (DVR) systems , where the user equipment is usually not as powerful as the head-end equipment.

Several studies [301, 302, 303, 304] compare the performance and suitability of JPEG2000 with respect to H.264/AVC when working with high-definition and high-quality video content, trying also to determine the applications (as digital cinema and archiving) and the benefits of working in

intra mode or/and visually lossless coding.

For example, in [301] an experimental study was performed with H.264/AVC and JPEG2000 in order to determine the benefits of using inter frame encoding versus intra frame encoding for digital cinema. Their results draw that the coding efficiency advantages of inter frame coding are significantly reduced for film content at the data rates and quality levels required by digital cinema. This indicates that the benefit of inter frame coding is questionable, because it is computationally much more complex, creates data access complexity due to the dependencies among frames, and in general, demands much more resources. For lower resolutions, their experiments confirm that inter frame coding was more efficient than intra frame coding. These results provide justification for using JPEG2000, or other intra frame coding methods, for coding digital cinema or high-quality/high-definition content. These studies use PSNR as distortion metric in their comparisons; in ours we will use VIF QAM.

So, for the applications mentioned above, a very interesting option to encode high-quality/high-definition video content is the use of intra coding systems, since they:

- Efficiently exploit the spatial redundancies of each video sequence frame.
- Exhibit reduced complexity in the design of the encoding/decoding engines.
- Achieve fast random access capability by decoding only the selected frame.
- Have great error resilience behavior by limiting error propagation to the frame boundaries.
- Are easily portable to parallel processing architectures, i.e., multicore CPUs .
- Have low coding/decoding delays, which it is of special interest for real-time applications.

So, we propose the use of the PETW encoder as perceptual intra encoder suited for high-quality/high-definition applications, which is able to perform very fast encoding (and decoding) with low demands of computational resources (processing power, and memory).

Now we will provide the results of the PETW quantizer when running for video sequences in intra mode. For this task, we have designed a Motion-PETW (M-PETW) version of the encoder that loads the video sequence in the

Table 3.13: Frame size, frame rate, and number of frames for the used sequences.

Sequence	Frame Rate	Frame Sizes	Frame Num
Foreman	30	QCIF (176x144) CIF (352x288)	300
Container	30	QCIF (176x144) CIF (352x288)	300
Hall	30	QCIF (176x144) CIF (352x288)	300
News	30	QCIF (176x144) CIF (352x288)	300
Mobile	30	ITU (720x576)	40
Station2	25	HD (1920x1024)	313
Pedestrian area	25	HD (1920x1024)	375
Ducks take off	50	HD (1920x1024)	130

YUV420 format and submits each frame to the core image encoder to obtain the perceptually enhanced frame.

The M-PETW receives the quantization step, $\Delta 1$ of Equation 3.39, as a parameter and encodes and decodes the whole sequence producing a single output perceptually enhanced video sequence in the YUV format. So, it provides the final bit rate for the desired quantization step. The VIF values for each frame are obtained independently in a batch process, with the final VIF value for the sequence the averaged quality values for all frames.

We have compared our M-PETW proposal with the following encoders in terms of R/D performance, coding delay, and memory consumption. All evaluated encoders have been tested on an Intel Pentium Core 2 CPU at 1.8 GHz with 6GB of RAM memory, employing several well-known video sequences with different formats - see Table 3.13 to see the characteristics of the used sequences where the frame size, frame rate, and number of frames are specified.

- Motion-JPEG2000 (Jasper 1.701.0)
- Motion-SPIHT (Spiht 8.01)
- X.264/Intra (FFmpeg version SVN-r25117, profile High, level 4.0)
- H.264/AVC/Intra (High-10, JM16.1 and JM18.1)

Although for a specific rate, a perceptually enhanced version provides in general higher VIF values than a non perceptually enhanced one, our interest is not to determine which encoder is the best but to measure how much bit rate in average can be saved when using a perceptually enhanced version, in particular our PETW versus non perceptually enhanced codecs.

So, first we will provide the results between the M-PETW and the M-LTW (the original non perceptually enhanced version) in order to determine the amount of rate that could be saved only by the inclusion of perceptual techniques in the same encoder. Later, we will test the M-PETW with the rest of the cited encoders.

To provide the results of the bit rate savings, we have use the same method as in section 3.3.1. We obtain 5 real Rate/VIF points evenly distributed along the rate axis, i.e., five points of real data. With this data, we use equations 3.29 and 3.30 to perform the curve fitting process so we can apply the Bjontegaard method to integrate over the VIF axis those points obtained with the M-PETW encoder. We have set the following perceptual quality ranges for the integration limits:

- Visually Lossless: $VIF > 0.83$

  It represents the sub-threshold value, i.e., above this threshold there are no perceptual differences with the original frames.

- Excellent: $0.60 <= VIF <= 0.83$

  In this quality range we include those video frames with very high perceptual quality.

- Good: $0.30 <= VIF < 0.60$

  In this quality range we include those video frames with perceptual quality varying from good to acceptable.

- All: $0.30 <= VIF <= 0.83$

  It covers the whole range of perceptual qualities, from acceptable to the visually lossless threshold.

In Table 3.14, the performance results in terms of bit rate saving between the M-PETW and the M-LTW encoders are shown for all the sequences in our test set. The table shows the rate savings for each of the cited quality ranges. It also provides the average value for each frame size. Values in bold type represent the maximum value for each quality level and frame size.

The bit rate gain at the *Visually Lossless* threshold, which is set at the 0.83 VIF value, is determined as the difference of rate exactly at this point, as for higher VIF values no perceptual differences are noticed. Although over this limit, two R/D curves diverge, or cross, the highest rate difference for the same perceptual quality is the one fixed at this limit.

Table 3.14: M-PETW versus M-LTW performance. Bit rate savings percentages for each quality range. Values for individual sequences and average for each frame size.

M-PETW vs. M-LTW	Visually Lossless	Excellent	Good	All
		QCIF (176x144)		
Foreman	8.55%	**10.10%**	**10.35%**	**10.19%**
Container	**10.63%**	9.65%	6.62%	8.50%
Hall	5.79%	5.49%	4.91%	5.27%
News	5.24%	4.76%	3.67%	4.34%
Seq. Average	7.55%	7.50%	6.39%	7.07%
		CIF (352x288)		
Foreman	**10.14%**	**12.14%**	**13.59%**	**11.63%**
Container	6.22%	7.53%	6.93%	6.75%
Hall	1.46%	3.27%	4.31%	2.75%
News	2.07%	3.82%	4.67%	3.27%
Seq. Average	4.97%	6.69%	7.37%	6.10%
		ITU (720x576)		
Mobile	**10.05%**	**8.34%**	**4.93%**	**7.99%**
		HD (1920x1024)		
Station2	4.53%	4.21%	2.21%	3.58%
Pedestrian area	7.31%	6.23%	4.81%	6.43%
Ducks take off	**16.22%**	**14.05%**	**5.54%**	**12.45%**
Seq. Average	9.35%	8.17%	4.19%	7.49%

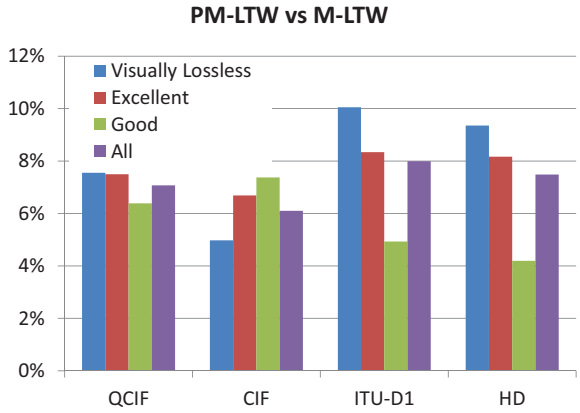


Figure 3.44: Performace comparison between M-PETW and M-LTW. Average bit rate savings for each frame size and quality segment.

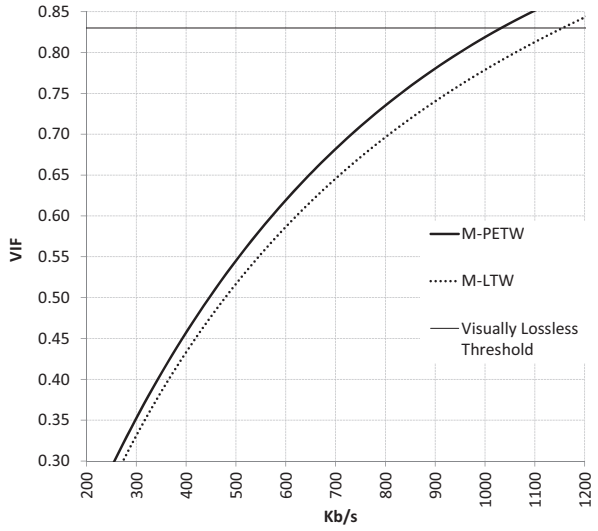


Figure 3.45: Rate distortion behavior comparison beteween M-PETW and M-LTW for the Container QCIF sequence.

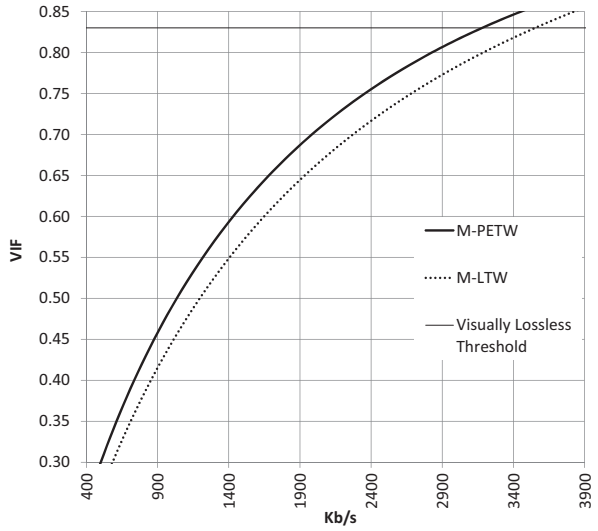


Figure 3.46: Rate distortion behavior comparison beteween M-PETW and M-LTW for the Foreman CIF sequence.

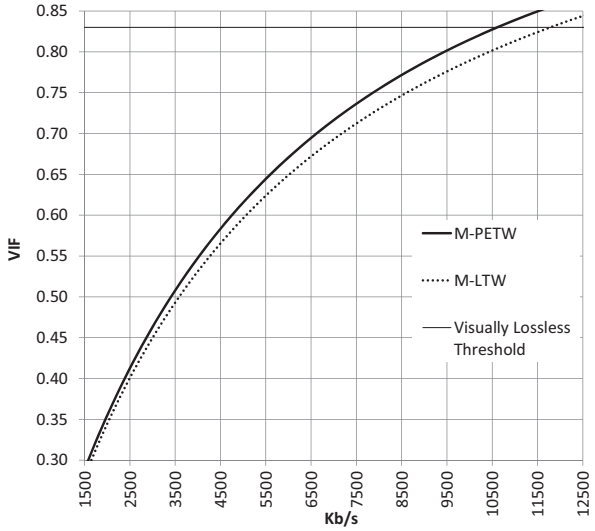


Figure 3.47: Rate distortion behavior comparison beteween M-PETW and M-LTW for the Mobile ITU-D1 sequence.

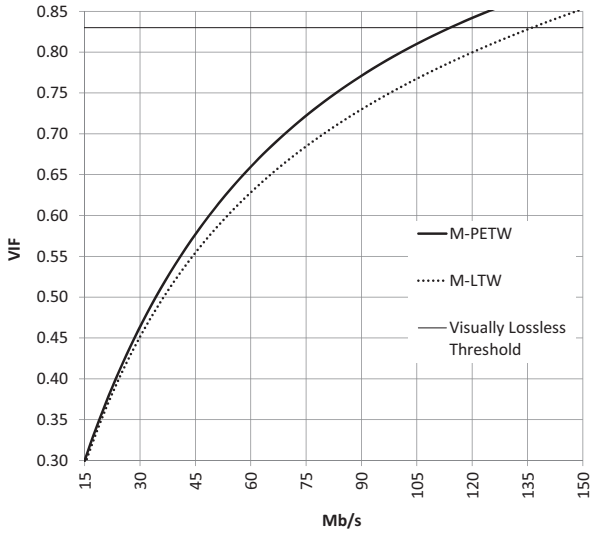


Figure 3.48: Rate distortion behavior comparison beteween M-PETW and M-LTW for the Ducks take off HD sequence.

For the QCIF frame size, the best gain is obtained at the *Visually Lossless* threshold for the Container sequence, with 10.14% of bit rate gain, whereas the average gain in this frame size is 7.55%. At the *Excellent* quality level, and for the QCIF size, the average gain is 7.50% being Foreman the best performing sequence with 10.10% of gain. Also, the Foreman sequence for this frame size is the best performing one for *Good*, and *All* quality levels, with approximately the same gains, being the average gain for these quality levels 6.39% and 7.07%, respectively.

For the CIF frame size, the best results are obtained again by the Foreman sequence providing a maximum gain of 10.14%, 12.14%, 13.59%, and 11.63% for the *Visually Lossless*, *Excellent*, *Good*, and *All* quality levels, respectively, and in the same order the average values are 10.05%, 8.34%, 4.93%, and 7.99%. The gain for the Foreman sequence in the *Good* quality segment is the highest of the test set in that segment.

For the HD frame size, the highest values are obtained for the Ducks take off sequence being also the maximum values for the *Visually Lossless*, *Excellent*, and *All* quality levels in the entire test set, these gains are 16.22%, 14.05%, and 12.45%, respectively.

The average gains for all the quality levels and frame sizes are summarized in Figure 3.44, and in figures from 3.45 to 3.48, the R/D comparison between M-PETW and M-LTW for several sequences are shown. In these figures, the visually lossless threshold is also represented in order to help to detect how much the rate gain is at that threshold.

In Table 3.15, the averaged results of comparing the M-PETW versus the rest of the encoders are shown. These are the average values for the different sequences at the corresponding frame sizes and quality levels. The maximum average gain of M-PETW versus all the encoders is produced at the *Visually Lossless* threshold for the HD frame size except when comparing with X.264. With that encoder, the best gain is obtained in the *Good* quality lelvel. The best averaged bit rate gains are 10.16%, 22.09%, 11.40%, 23.11%, and 10.69% in the comparisons with M-JASPER, M-SPIHT, KKDU, X.264, and H.264, respectively.

The negative values in Table 3.15 refer to a loss of bit rate with the compared encoder. For example, in the comparison against H.264, the M-PETW encoder only gets better results for the ITU-D1 frame size for the Visually Lossless threshold, and for all the quality levels in the HD frame size. For lower resolutions, H.264 obtains better results on average. The same happens when comparing with X.264, but in this case M-PETW obtains worse results only for small frame sizes (QCIF and CIF).

Table 3.15: Comparison results of the M-PETW encoder versus other encoders. Average bit rate savings values for each frame size and quality range.

M-PETW vs. M-JASPER	QCIF	CIF	ITU	HD
Visually Lossless	6.77%	2.38%	8.81%	**10.16%**
Excellent	8.15%	4.23%	8.29%	8.62%
Good	5.24%	6.66%	8.87%	7.23%
All	6.85%	5.14%	8.48%	8.19%

M-PETW vs. M-SPIHT	QCIF	CIF	ITU	HD
Visually Lossless	11.09%	11.08%	14.89%	**22.09%**
Excellent	11.25%	11.24%	13.56%	18.62%
Good	4.65%	9.74%	12.00%	12.62%
All	8.83%	10.73%	13.07%	16.81%

M-PETW vs. KKDU	QCIF	CIF	ITU	HD
Visually Lossless	10.08%	7.51%	11.07%	**11.40%**
Excellent	11.19%	8.13%	10.70%	7.77%
Good	10.97%	8.26%	11.31%	2.31%
All	10.91%	8.11%	10.89%	6.11%

M-PETW vs. X.264	QCIF	CIF	ITU	HD
Visually Lossless	-1.95%	-2.95%	13.08%	15.32%
Excellent	-3.72%	-3.67%	12.47%	18.39%
Good	-12.81%	-7.15%	12.76%	**23.11%**
All	-7.05%	-4.89%	12.56%	19.99%

M-PETW vs. H.264	QCIF	CIF	ITU	HD
Visually Lossless	-8.13%	-7.25%	3.10%	**10.69%**
Excellent	-12.99%	-11.41%	-5.70%	8.32%
Good	-30.05%	-22.79%	-13.73%	2.23%
All	-18.54%	-15.19%	-8.15%	6.53%

In figures 3.49 to 3.52, we can see some of the R/D plots for these comparisons, and in Table 3.16 the maximum gain obtained for each quality level and frame size is shown. We can see gains up to 28.01% in the *Good* quality level when comparing with X.264, 22.09% in the *Visually Lossless* threshold when comparing with M-SPIHT, and even up to 12.85% when comparing with H.264.

These results show that higher gains are obtained as the video resolution increases (see Figure 3.53), when comparing with DCT based encoders as X.264, and H.264, and also, but with a lower slope, this trend is present while comparing with M-SPIHT. In the comparison with the JPEG2000 derived encoders (M-JASPER and KKDU) this fact has not been met, as the gain is

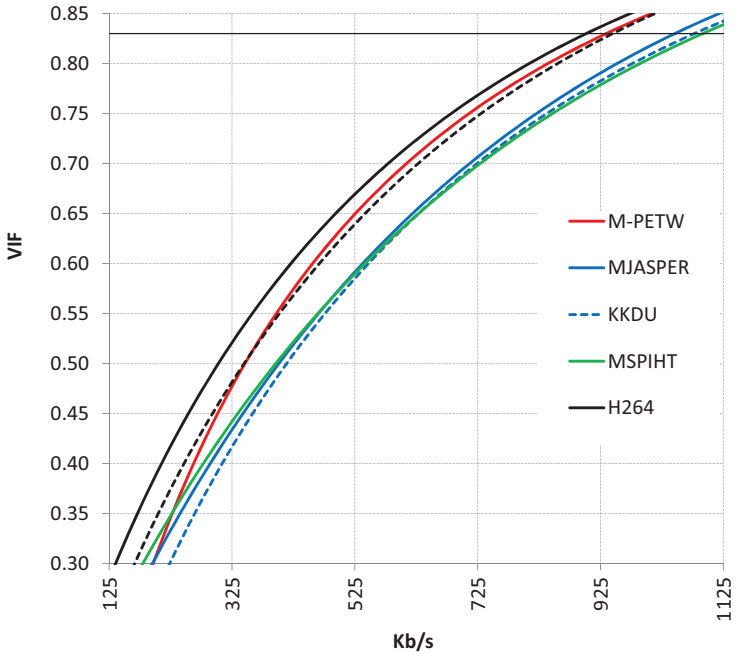


Figure 3.49: Rate distortion behavior of the different encoders for the Foreman QCIF sequence.

almost constant. Nevertheless, this comparison still provides an average gain of 8.62%, and 7.77% in the *Excellent* quality range with respect to M-JASPER and KKDU. If we take the *All* quality level, those differences are 8.19% and 6.11% with respect to the same encoders.

Now, we will proceed to compare some of the codecs under test in terms of coding delay and memory requirements.

Figure 3.54 shows the coding speed in frames per second obtained by the different encoders being evaluated. As shown, M-PETW outperforms the rest of the codecs for any sequence frame resolution. For the highest resolution, M-PETW is 1.08 times as fast as M-SPIHT, 2.22 times as fast as M-JASPER, 2.30 times as fast as X.264, and 28.09 times as fast as H.264/AVC.

It is important to notice that the current implementation of our codec is not optimized in any sense. While comparing with M-JPEG2000 using KKDU, execution times of M-PETW are faster only for the QCIF frame resolution. The reason is that KKDU is fully optimized including multi-thread and multicore hardware capabilities, processor intrinsics like MMX/SSE/SSE2/SIMD and fast multicomponent transform. Therefore

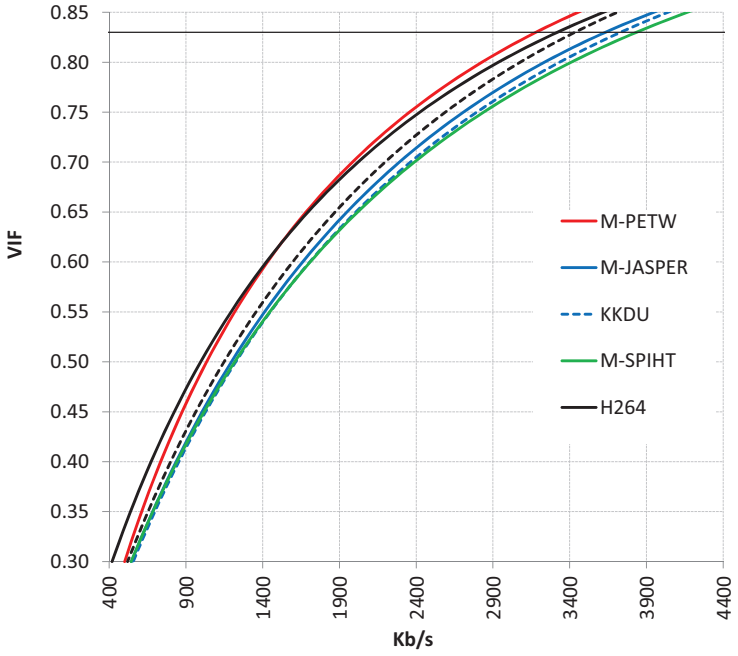


Figure 3.50: Rate distortion behavior of the different encoders for the Foreman CIF sequence.

KKDU outperforms M-PETW in coding time, processing up to 102.13 fps (frames per second) in CIF resolution, 42.43 fps in ITU-D1 resolution and 14.0 fps in HD resolution.

Regarding memory requirements, in Figure 3.55 we can see the maximum amount of memory (in Mbytes) required for each encoder and resolution. As can be seen, M-PETW requires nearly 4 times less memory resources as M-SPIHT, M-JASPER, and X.264, and up to 40 times less memory than H.264/AVC.

3.4.3 Variable dead zone optimization

Our objective in this section is focused into the impact of the dead zone size on the R/D coding performance. So, we will analyze how different dead zone sizes affect to the VIF R/D performance, and then we will propose a way to estimate the dead zone size that maximizes the VIF R/D performance and therefore, the perceptual quality.

In [299, 274], Marcellin et al. showed the influence of the dead zone size in

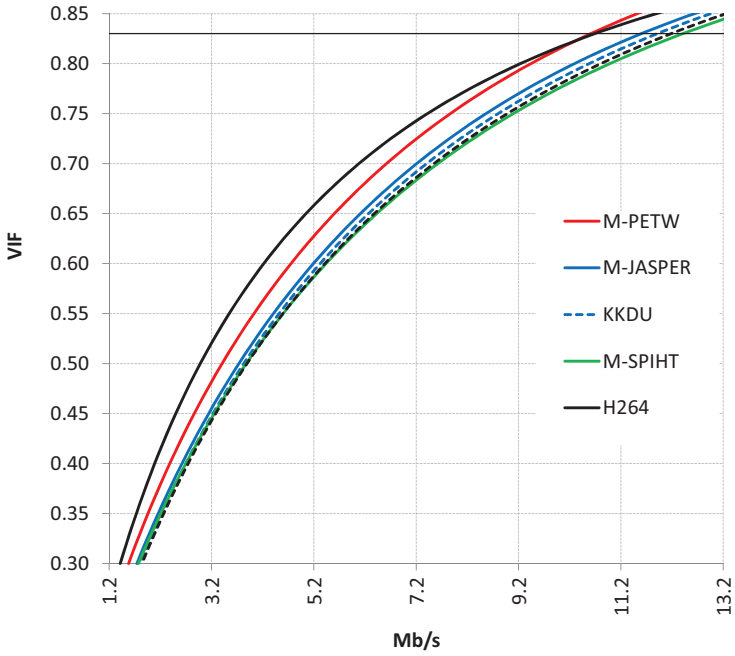


Figure 3.51: Rate distortion behavior of the different encoders for the Mobile ITU-D1 sequence.

the R/D performance. Their study was done in terms of PSNR. Our studies are oriented, by contrast, to optimize the R/D behavior in terms of the VIF QAM, and hence to determine which is the influence of the dead zone size over the perceptual quality, and not over the PSNR as in previous studies.

Also, the LTW authors exposed in [226] that experimental tests have revealed that the use of the constant K in the quantization formulation (see Section 3.4.1), whose objective is to take some values out of the dead zone, i.e., narrowing it, can increase the PSNR R/D performance of the LTW. Their studies suggested that a value of $K = 1$ is appropriate for most source images. This value produces, as studied before, a dead zone size of 1.25Δ, being this size the same for all images.

In [305], Ström made an experiment to determine how large the dead zone should be for optimal performance, and how much quality could be gained when in a DWT encoder the USQ is substituted with a USDZQ. Their study was done also in terms of R/D performance with the PSNR as quality metric. They used only one image, and their results found that for this test image the optimal size was about 1.9Δ, which finally provides a quality increase of 0.5 dBs for that image.

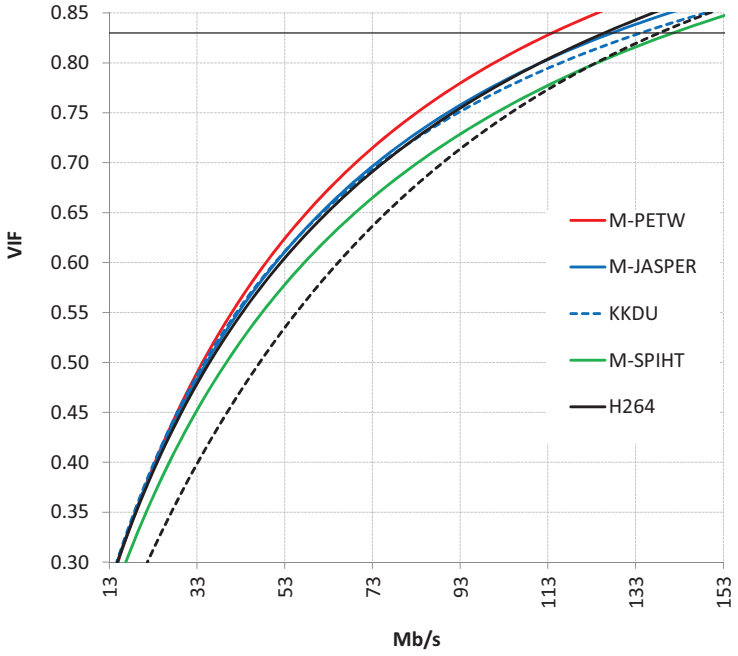


Figure 3.52: Rate distortion behavior of the different encoders for the Ducks take off HD sequence.

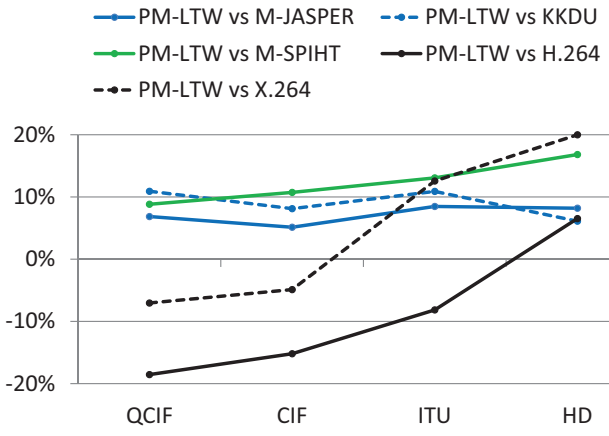


Figure 3.53: Bit rate savings in relationship with the frame resolution.

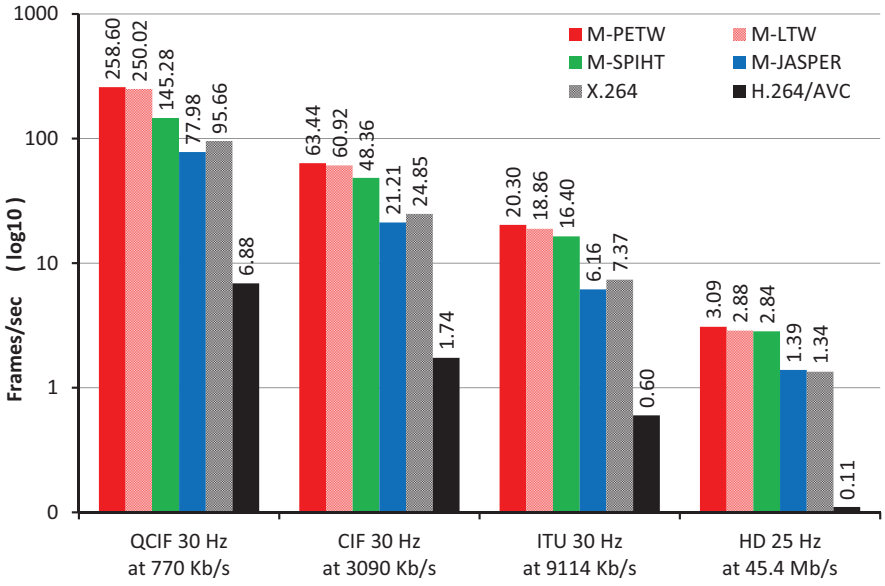


Figure 3.54: Encoder frame rate at different sequence resolutions.

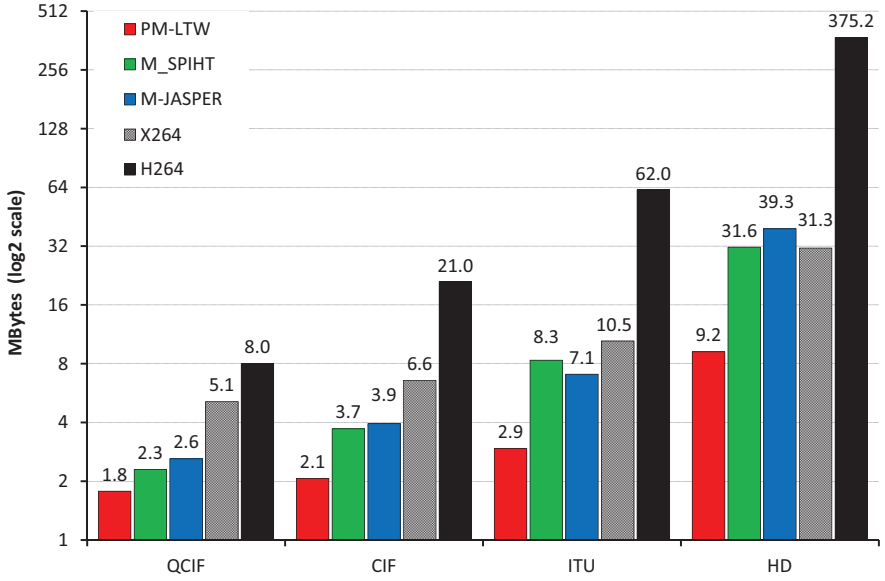


Figure 3.55: Memory requirements for different video formats.

Table 3.16: Comparison results of the M-PETW encoder versus other encoders. Maximum bit rate savings values for each frame size and quality range.

M-PETW vs. M-JASPER	QCIF	CIF	ITU	HD
Visually Lossless	10.28%	12.45%	8.81%	11.13%
Excellent	**13.44%**	13.09%	8.29%	13.20%
Good	11.45%	12.71%	8.87%	15.31%
All	11.87%	12.96%	8.48%	13.87%

M-PETW vs. M-SPIHT	QCIF	CIF	ITU	HD
Visually Lossless	14.13%	17.07%	14.89%	**22.09%**
Excellent	15.81%	16.89%	13.56%	18.62%
Good	9.38%	13.97%	12.00%	12.62%
All	13.47%	15.92%	13.07%	16.81%

M-PETW vs. KKDU	QCIF	CIF	ITU	HD
Visually Lossless	13.55%	14.63%	11.07%	**15.75%**
Excellent	15.20%	15.65%	10.70%	10.83%
Good	15.63%	14.69%	11.31%	6.65%
All	14.61%	15.33%	10.89%	9.57%

M-PETW vs. X.264	QCIF	CIF	ITU	HD
Visually Lossless	2.13%	7.14%	13.08%	17.31%
Excellent	2.43%	9.07%	12.47%	21.10%
Good	-3.17%	8.96%	12.76%	**28.01%**
All	0.32%	9.03%	12.56%	23.48%

M-PETW vs. H.264	QCIF	CIF	ITU	HD
Visually Lossless	-1.86%	4.38%	3.10%	**12.85%**
Excellent	-4.96%	2.36%	-5.70%	9.71%
Good	-18.76%	-6.00%	-13.73%	8.51%
All	-8.11%	-0.32%	-8.15%	9.32%

As with the Ström experiment, we will also use a DWT based encoder, in this case our PETW proposal that implements a UVDZQ. We will use VIF as distortion metric because the optimum dead zone should be calculated taking into account the perceptual elevation produced by the PWM.

In a first experiment, we use several well-known images such as Lena, Mandrill, Barbara, and Boat, and also the whole image Kodak set. For those images, and following the same methodology as in the Ström experiment, we fix the target bit rate, in this case to 0.4 bpp. For that target bit rate, we compress the image varying the dead zone size by means of the ξ (Xi) parameter of equations 3.46 and 3.47 and the resulting images were compared against the original one using the VIF metric.

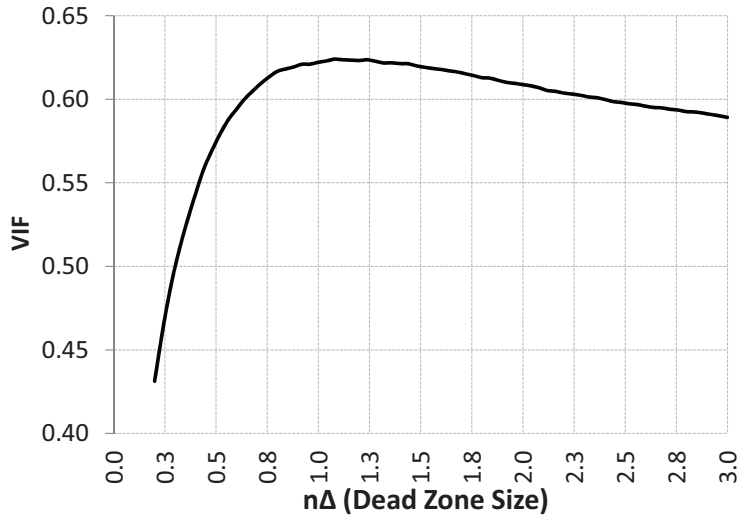


Figure 3.56: VIF variation for image 07 from the Kokak set at 0.4 bpp when the dead zone size varies from 0.2Δ to 3.0Δ.

This experiment obtains a dead zone size that is optimum for the target bit rate, but it does not prove if the same dead zone size is optimum for the rest of bit rates in the range where the R/D behavior is analyzed. Furthermore, in order to fix the target bit rate, once the dead zone size has been fixed, we must change the step size to reach the target bit rate, and therefore the experiment changes two variables, not only the dead zone size. Nevertheless, the experiment highlights the existence of an optimum dead zone size that maximizes the perceptual quality at a desired bit rate, and shows the importance and relationship of the dead zone size with the final perceptual quality.

Depending on the image, content the VIF gain varies, but for example as shown in Figure 3.56, which corresponds to image 07 from Kodak set, we vary the dead zone size in the range from 0.2Δ to 3.0Δ and for a fixed target bit rate of 0.4 bpp we obtain the best VIF value when $\xi = 0.460$ that corresponds to a dead zone size of 1.08Δ. Using this dead zone size instead of 0.2Δ, moves the image classification from the *Good* quality range into the *Excellent* one according to the classification made in previous sections.

In Table 3.17, the results for this experiment applied to the whole Kodak image set are shown. Images are ordered in ascending order of maximum VIF gain. For this experiment, the ξ parameter varies in the range $-0.500 <= \xi <= 1$, which corresponds to a dead zone variation from 3Δ to 0Δ respectively, and the bit rate is fixed at 0.4 bpp for all images. Column *Max. VIF gain* is the

Table 3.17: Maximum VIF gains for varying dead zone size at 0.4 bpp for the whole Kodak image set.

Image	Max. VIF gain	DZ Size	ξ
13	0.02	1.12	0.440
08	0.02	1.24	0.380
05	0.04	1.40	0.300
01	0.04	1.12	0.440
20	0.07	1.04	0.480
14	0.07	1.16	0.420
02	0.08	0.92	0.540
12	0.08	0.72	0.640
18	0.09	1.28	0.360
11	0.10	1.00	0.500
16	0.10	0.76	0.620
19	0.11	0.88	0.560
09	0.12	0.88	0.560
04	0.12	0.96	0.520
23	0.12	1.00	0.500
22	0.13	1.08	0.460
15	0.13	1.08	0.460
03	0.13	0.84	0.580
10	0.13	0.88	0.560
21	0.15	1.12	0.440
17	0.17	1.04	0.480
06	0.19	1.08	0.460
07	0.19	1.08	0.460

maximum VIF gain obtained for each image in the specified ξ range, *DZ Size* column is the dead zone size for the maximum VIF gain, and the ξ column is the corresponding ξ value for this optimum dead zone size.

As the study was done for a fixed rate, 0.4 bpp, the next thing to prove is if with the optimum ξ at this rate for an image, the R/D behavior for the whole rate range behaves better than or at least equal to the one obtained with the standard PETW, i.e., when the equivalent dead zone size is used (1.25Δ for $\xi = 0.375$). The PETW equivalent dead zone size is not necessary the best one, as the study performed with the LTW was done in terms of PSNR and without the perceptual elevation of the coefficients, but in some cases it could be optimum or near to it.

In Figure 3.57, the R/D behavior is shown for different ξ values. It is clearly shown that a change in the dead zone size produces different R/D behavior. The one with best performance corresponds to $\xi = 0.500$, which is not the optimum obtained in the previous experiment ($\xi = 0.450$), although it is not far from it.

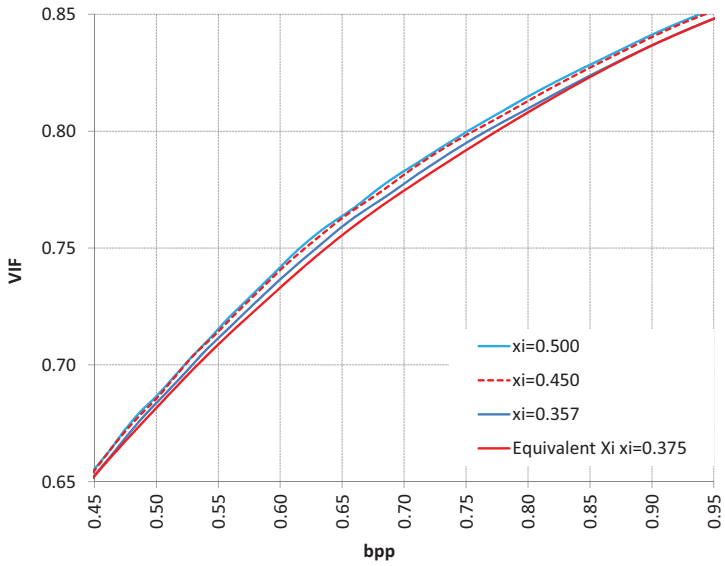


Figure 3.57: VIF variation for image 07 of the Kokak set, at 0.4 bpp when the dead zone size varies from 1Δ to 3.0Δ

As mentioned before, this is because the experiment obtains the optimum for a specific bit rate (0.4 bpp), and the step size changes besides, in the different executions in order to fix it to the target bit rate.

Nevertheless, as shown from previous results, it is worthwhile to fix the dead zone size to the optimum for each image. The optimum should be the one that maximizes the bit rate gain for a specific quality range, but the most difficult task is to automatically obtain this optimum for each image while encoding it, i.e., obtaining adaptively the optimum dead zone size for each image. Additionally, we should take special care to perform the optimum estimation without increasing the computational cost of the encoder.

We performed another experiment to obtain the optimum ξ parameter from a R/D point of view. In this case we choose five step sizes, i.e., five values for the Q parameter of the PETW that produce five rates evenly spaced in the bit rate range. With the real VIF/rate values we use Equation 3.29 to estimate the VIF R/D curve for those points. We produce 101 estimated curves for each image, one curve for each ξ value in the range $-0.500 <= \xi <= 1$ chosen in increments of 0.010 units. We call these curves *Xi Curves*. Doing so, the only parameter that changes in the PETW encoder is the ξ one, as the step sizes for each of the curves are the same.

Then, for each of the *Xi Curves*, we obtain the bit rate gain or loss with the

Bjontegaard method as in previous sections. We compare the gain or loss of each curve with the one obtained with the reference curve. The reference curve is the one obtained with $\xi = 0.375$ that equals the 1.25Δ dead zone size of the S-LTW. We used the whole image Kodak set to obtain the optimum ξ value for each image, we call it *best xi value*, i.e., the value that maximizes the bit rate gain with respect to the reference R/D curve in the VIF range from 0.30 to 1.0 VIF units.

Table 3.18 shows the best xi values for the Kodak set images. As said before, the objective is not to perform these calculations for each image. We search for one value that could be calculated on the fly or used as a well-working global value.

One way to avoid the task of calculating the best xi for every image is to obtain a unique value that is sub-optimum for the image. One candidate value to use is the mean or median ξ value of column *best xi* in Table 3.18. The mean value is 0.077 and the median value is 0.120. In Table 3.18, the *Median Err.* and *Mean Err.* columns show the estimation error between these values and the optimum xi value from the *best xi* column. The last row shows the mean error for each of these estimated values. Although for some images these estimated xi values produce practically the same VIF R/D curve than the one obtained with the best xi value, none of them is a good approximation because for some images the R/D curve is below the reference one.

So the objective is to find another estimated xi value that is able to minimize this averaged error. We then searched for a correlation between the best xi values and some statistical value or metric obtained directly from the image, or from the wavelet coefficients before the quantization is performed, as the ξ parameter must be known at this point.

A first option is to use the SD of the wavelet coefficients, but as shown in Figure 3.58, where the SD of the LL subband is used, there is no appreciable correlation between the best xi values from Table 3.18 and the SD for each image, shown on the horizontal axis.

So we proceeded to search for some entropy measures, which we call *estimators*, that are able to estimate the bpp used for each image, producing an estimation of the bpp value, E_{bpp}. These estimators were implemented in the PETW before the quantization stage, and therefore before the encoding stage. We implemented three estimators:

- *Coefficient Entropy*: This is the zero order entropy obtained directly from the wavelet coefficients after transform. This is a generic measure that does not depend on the encoder.

Table 3.18: Best xi values for the Kodak set images.

Image	Best Xi	Median Err.	Mean Err.
1	0.340	0.220	0.263
2	-0.360	0.480	0.437
3	-0.190	0.310	0.267
4	-0.130	0.250	0.207
5	0.320	0.200	0.243
6	0.270	0.150	0.193
7	0.120	0.000	0.043
8	0.250	0.130	0.173
9	-0.030	0.150	0.107
10	-0.070	0.190	0.147
11	0.160	0.040	0.083
12	-0.220	0.340	0.297
13	0.410	0.290	0.333
14	0.220	0.100	0.143
15	-0.100	0.220	0.177
16	0.050	0.070	0.027
17	0.080	0.040	0.003
18	0.260	0.140	0.183
19	0.170	0.050	0.093
20	0.080	0.040	0.003
21	0.270	0.150	0.193
22	0.130	0.010	0.053
23	-0.250	0.370	0.327
	Avg.	0.171	0.174

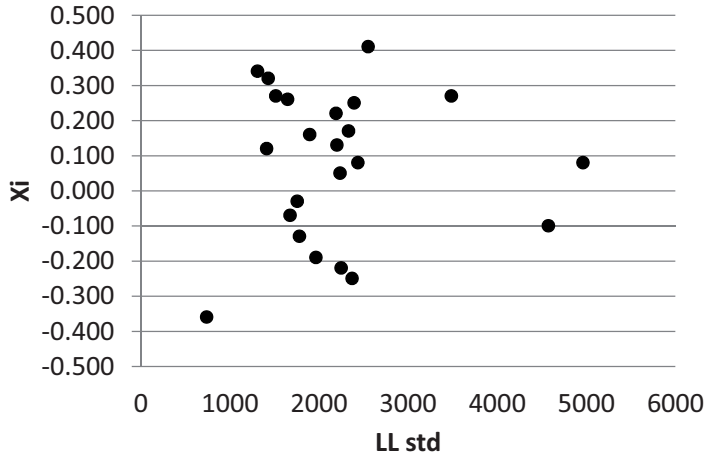


Figure 3.58: Dispersion plot for the *best xi* vs. LL std for images in the Kodak set.

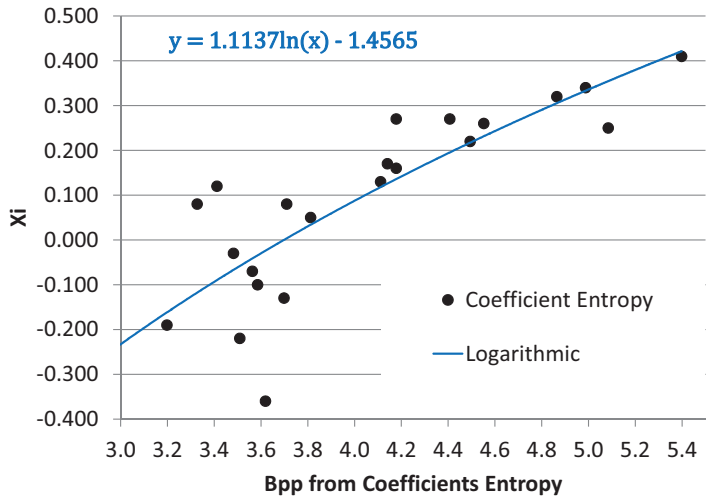


Figure 3.59: Scatter plot for the Best Xis vs. E_{bpp} obtained with the *Coefficient Entropy* estimator for images in the Kodak set. Logarithmic fitting equation is also shown.

- *Symbols Entropy*: This is the zero order entropy of the PETW symbol map used in the encoding process. This measure depends strictly on the PETW encoder as the symbols will be used by the encoding algorithm.
- *PETW Bpp*: This is an entropy estimation that uses the E_{bpp} produced by *Symbols Entropy* plus the real amount of bpp used for the raw bits of each of the coefficients. In order to determine the real bits needed for each coefficient, a dead zone size and a step size must be fixed. We use a dead zone size equivalent to the one used by the rate control stage in the S-LTW, which uses a $rplanes = 2$ with no further quantization. This estimator is also dependent on the PETW.

Once we have the E_{bpp} from each estimator, we use a scatter plot to see if there is some correlation between the E_{bpp} and the optimum xi for each image in the Kodak set, see Table 3.18. In figures 3.59 to 3.61, we see these scatter plots, where a correlation is shown.

Figures 3.59 and 3.60 also show the best fitting equation (logarithmic in both cases), that is used to estimate the best xi values for a desired bit rate. In Figure 3.61, a polynomial fitting is shown instead.

As mentioned before, the objective is then to find which one of the fitting equations produce less averaged error while estimating the best xi value.

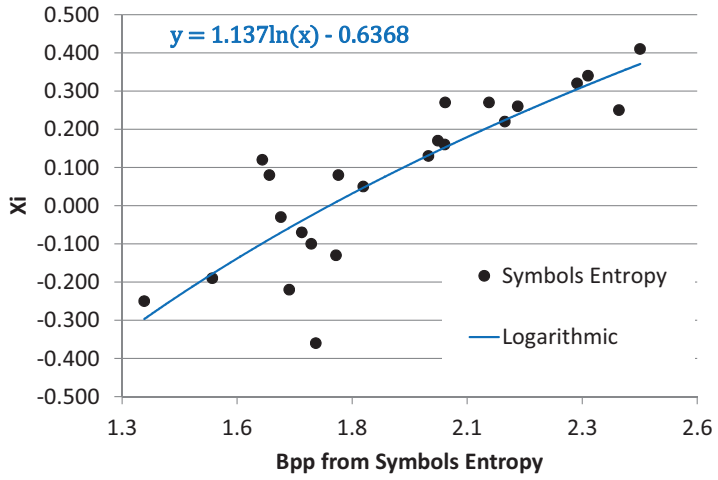


Figure 3.60: Scatter plot for the Best Xis vs. E_{bpp} obtained with the *Symbols Entropy* estimator for images in the Kodak set. Logarithmic fitting equation is also shown.

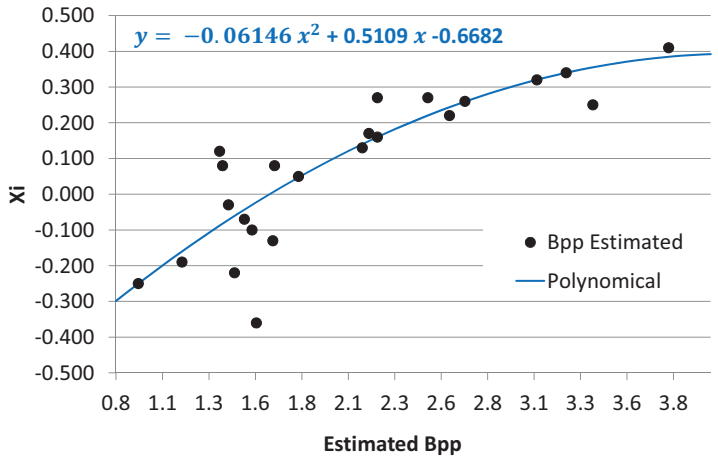


Figure 3.61: Scatter plot for the Best Xis vs. E_{bpp} obtained with the *PETW Bpp* estimator for images in the Kodak set. Polynomial fitting equation is also shown.

Table 3.19 shows the results for the *Coefficient Entropy* and the *Symbols Entropy* estimators. Columns are the Kodak set image number, the best ξ from the previous experiment, the E_{bpp} obtained with the corresponding estimator, the $E\xi$ (estimated Xi) value obtained with the fitting equation, and the error with respect to the optimum ξ for the fitting equation. In the last row, the average error is also shown for each fitting equation. Table 3.20 shows the same information, but related to the *PETW Bpp* estimator.

$$E\xi = -0.06146E_{bpp}^2 + 0.5109E_{bpp} - 0.6682 \tag{3.48}$$

The worst results are obtained with the *Coefficient Entropy*, which is the only encoder-independent estimator. However, a much better estimation is obtained than using the Mean or the Median of the optimum xis. Therefore, this estimator could be used in any wavelet based encoder that uses a dead zone quantizer, like for example JPEG2000, although some adaptations and more experiments must be done.

In the case of the PETW-dependent estimators, the best results are obtained with the *PETW Bpp*, which is also the better of the three estimators that we have implemented. It obtains an average error of 0.069ξ. The polynomial fitting equation used in *PETW Bpp* is also shown in Equation 3.48, where $E\xi$ stands for estimated Xi, and E_{bpp} is the estimated bpp obtained with it.

Once we have an equation to estimate the best ξ for a specific image, we will test it with other well-known images. In this case we will use also higher resolution images. The image test set includes the images refereed in Table 3.21.

As an example of the estimation performance of the *PETW Bpp* estimator, in figures 3.62 and 3.63 we show, for the Lena and Balloon images, respectively, three R/D curves in each figure. One curve for the PETW with the *Equivalent Xi* value ($\xi = 0.375$), another with the *Optimum Xi* obtained with the aforementioned experiment (the one that uses the Bjontegaard method to determine which curve was the best), and the last curve, the one obtained with the *Estimated Xi* ($E\xi$ value). The curve for the *Optimum Xi*, and the curve for the *Estimated Xi* are practically the same, so in order to be able to see both curves at the same time, we have plotted the *Estimated Xi* curve with a thicker line.

The *PETW Bpp* estimator is quite precise as for most images the curves obtained with the optimum xi and the estimated one are practically the same. The average error between the $E\xi$ value and the optimum xi for the test images is 0.082ξ, which is equivalent to a deviation in the dead zone size of only 0.16Δ with respect to the optimum.

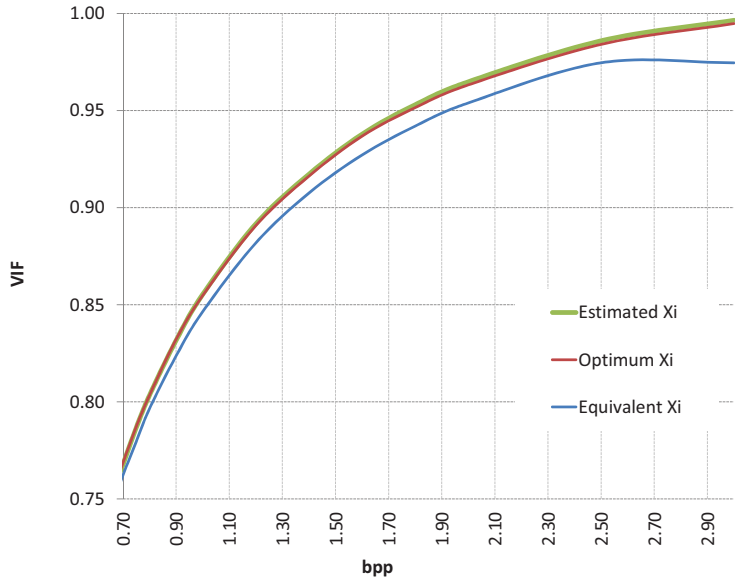


Figure 3.62: Lena: R/D curve comparison between the Optimum Xi, Estimated Xi, and Equivalent Xi values.

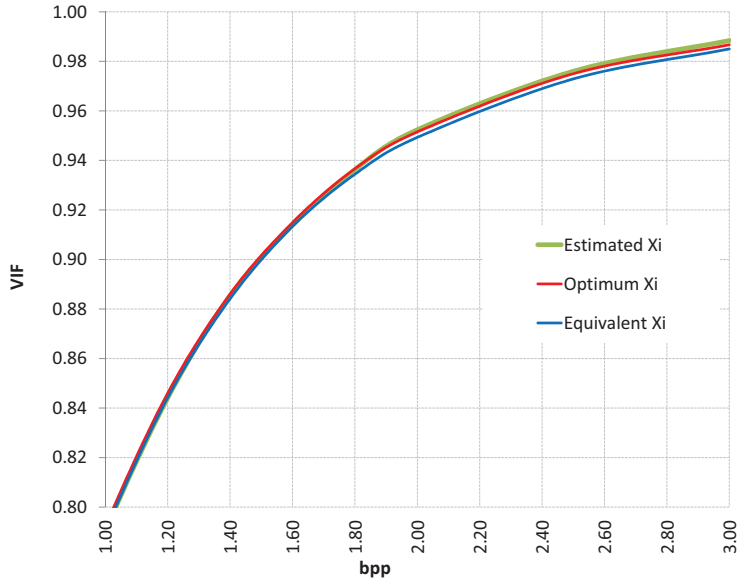


Figure 3.63: Balloon: R/D curve comparison between the Optimum Xi, Estimated Xi, and Equivalent Xi values.

Table 3.19: Results for the *Coefficient Entropy* and *Symbols Entropy* estimator. Image and average error for the fitting equations.

Image	Best Xi	Coefficient Entropy			Symbols Entropy		
		Est. bpp	Log. Xi	Log. Err.	Est. bpp	Log. Xi	Log. Err.
01	0.340	4.99	0.333	0.007	2.31	0.316	0.024
02	-0.360	3.62	-0.024	0.336	1.72	-0.020	0.340
03	-0.190	3.20	-0.162	0.028	1.50	-0.179	0.011
04	-0.130	3.70	0.000	0.130	1.77	0.009	0.139
05	0.320	4.86	0.305	0.015	2.29	0.305	0.015
06	0.270	4.41	0.195	0.075	2.10	0.205	0.065
07	0.120	3.41	-0.090	0.210	1.60	-0.099	0.219
08	0.250	5.08	0.354	0.104	2.38	0.349	0.099
09	-0.030	3.48	-0.067	0.037	1.65	-0.071	0.041
10	-0.070	3.56	-0.041	0.029	1.69	-0.040	0.030
11	0.160	4.18	0.136	0.024	2.00	0.152	0.008
12	-0.220	3.51	-0.058	0.162	1.66	-0.058	0.162
13	0.410	5.40	0.421	0.011	2.43	0.371	0.039
14	0.220	4.49	0.217	0.003	2.13	0.224	0.004
15	-0.100	3.59	-0.035	0.065	1.71	-0.026	0.074
16	0.050	3.81	0.034	0.016	1.82	0.047	0.003
17	0.080	3.71	0.004	0.076	1.77	0.012	0.068
18	0.260	4.55	0.231	0.029	2.16	0.239	0.021
19	0.170	4.14	0.126	0.044	1.99	0.144	0.026
20	0.080	3.33	-0.118	0.198	1.62	-0.088	0.168
21	0.270	4.18	0.136	0.134	2.00	0.153	0.117
22	0.130	4.11	0.118	0.012	1.97	0.132	0.002
23	-0.250	2.99	-0.237	0.013	1.35	-0.297	0.047
			Avg. Err.	0.076		Avg. Err.	0.075

In Table 3.22, the differences or deviations from the $E\xi$ to the equivalent xi (0.375) and their translation into dead zone size deviations are shown. For the test images, the table shows in the *Estimation* columns: $E\xi$ the estimated xi, and *DZ* the corresponding dead zone size, and in the *Deviation* columns: ξ the distance or deviation from the equivalent xi, and *DZ* the translation into dead zone size units. As shown, deviations over 0.300ξ units produce dead zone deviations higher than 0.5Δ with respect to the equivalent dead zone size.

This is a big deviation from the dead zone size used by the equivalent PETW. Depending on the image content, this dead zone deviation can have a higher impact on the R/D performance at lower compression rates as shown in figures 3.65, 3.64, 3.66, and 3.67, where the gains are shown for Woman, Zelda, Deer and Big Tree images. These images are the ones from Table 3.22 with bigger deviations of the estimated xi value. In that images, the previously used visually lossless threshold is also shown.

Table 3.20: Results for the *PETW Bpp* estimator. Image and average error for the fitting equation.

Image	Best Xi	Estimated Bpp		
		Est. bpp	Est. Xi	Poly. Err.
01	0.340	3.22	0.340	0.000
02	-0.360	1.56	-0.022	0.338
03	-0.190	1.16	-0.160	0.030
04	-0.130	1.64	0.006	0.136
05	0.320	3.06	0.320	0.000
06	0.270	2.48	0.221	0.049
07	0.120	1.36	-0.088	0.208
08	0.250	3.37	0.355	0.105
09	-0.030	1.41	-0.072	0.042
10	-0.070	1.49	-0.043	0.027
11	0.160	2.21	0.160	0.000
12	-0.220	1.44	-0.061	0.159
13	0.410	3.77	0.385	0.025
14	0.220	2.59	0.244	0.024
15	-0.100	1.53	-0.030	0.070
16	0.050	1.78	0.047	0.003
17	0.080	1.65	0.008	0.072
18	0.260	2.68	0.259	0.001
19	0.170	2.16	0.149	0.021
20	0.080	1.37	-0.082	0.162
21	0.270	2.21	0.160	0.110
22	0.130	2.13	0.140	0.010
23	-0.250	0.92	-0.250	0.000
			Avg. Err.	0.069

Table 3.21: Image set used in the variable dead zone experiments.

Images	Resolution
Lena Barbara Goldhill Boat Mandrill Balloon Horse Zelda	512 x 512
Cafe Bike Woman	2048 2560
Deer	3968 x 2560
Big_Tree	6016 x 4480

Table 3.22: ξ and dead zone deviations from the equivalent values.

	Estimation		Deviation	
Images	$E\xi$	DZ	ξ	DZ
Lena	-0.019	2.04 Δ	**0.394**	**0.79** Δ
Barbara	0.194	1.61 Δ	0.181	0.36 Δ
Goldhill	0.163	1.67 Δ	0.212	0.42 Δ
Boat	0.073	1.85 Δ	0.302	0.60 Δ
Mandrill	0.380	1.24 Δ	0.005	0.01 Δ
Balloon	0.145	1.71 Δ	0.230	0.46 Δ
Horse	0.292	1.42 Δ	0.083	0.17 Δ
Zelda	-0.151	2.30 Δ	**0.526**	**1.05** Δ
Cafe	0.035	1.93 Δ	0.340	0.68 Δ
Bike	0.144	1.71 Δ	0.231	0.46 Δ
Woman	0.116	1.77 Δ	0.259	0.52 Δ
Deer	-0.003	2.01 Δ	**0.378**	**0.76** Δ
Big_Tree	-0.184	2.37 Δ	**0.559**	**1.12** Δ

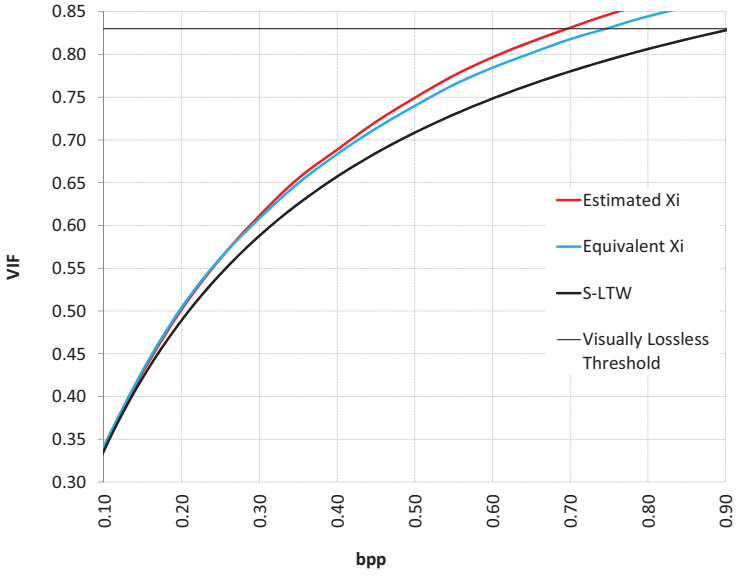


Figure 3.64: Zelda: R/D curve comparison between the Estimated Xi and Equivalent Xi values.

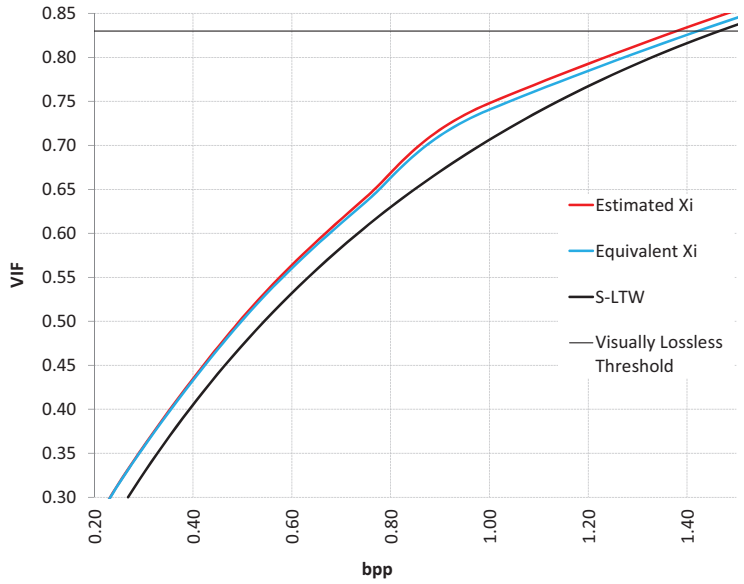


Figure 3.65: Woman: R/D curve comparison between the Estimated Xi and Equivalent Xi values.

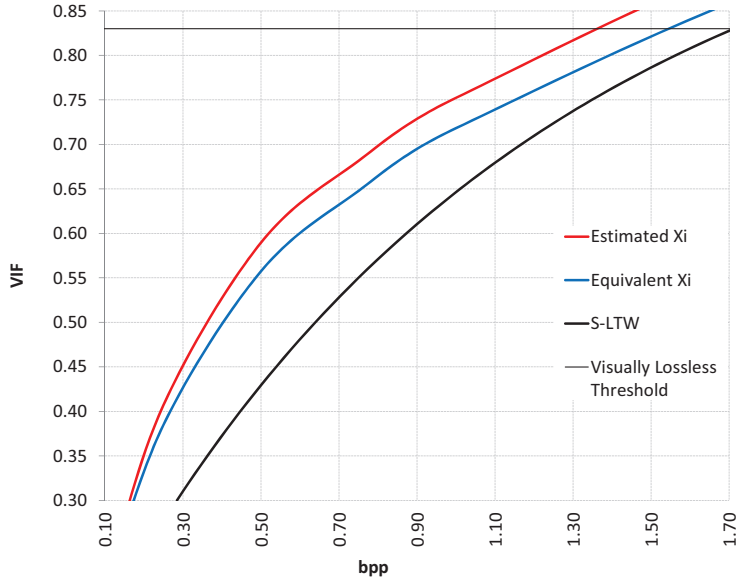


Figure 3.66: Deer: R/D curve comparison between the Estimated Xi and Equivalent Xi values.

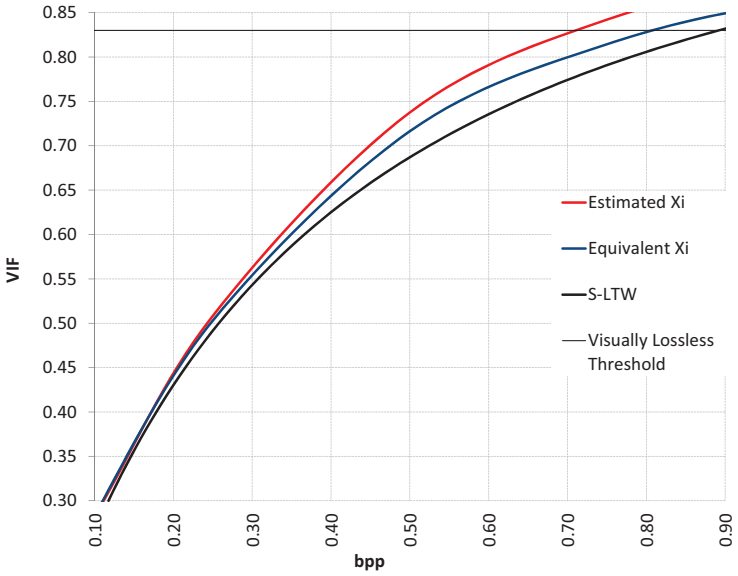


Figure 3.67: Big Tree: R/D curve comparison between the Estimated Xi and Equivalent Xi values.

Table 3.23: Additional % of bit rate gain/loss due to the use of the *PETW Bpp* estimator for the VL and E quality ranges.

Images	Visually Lossless		Excellent	
	% Add.	% Tot.	% Add.	% Tot.
Lena	**3.80%**	18.08%	**1.89%**	14.59%
Barbara	1.18%	12.35%	0.84%	14.32%
Goldhill	2.33%	9.46%	1.73%	11.12%
Boat	1.62%	8.29%	0.91%	7.79%
Mandrill	-0.07%	10.96%	-0.05%	13.83%
Balloon	0.17%	9.39%	-0.18%	9.14%
Horse	0.72%	10.55%	0.54%	12.58%
Zelda	**6.50%**	23.50%	**3.54%**	16.70%
Cafe	0.57%	8.87%	-0.63%	9.81%
Bike	1.91%	11.25%	0.24%	12.02%
Woman	2.95%	9.17%	2.09%	9.87%
Deer	**16.11%**	28.65%	**13.34%**	34.24%
Big Tree	**9.98%**	16.37%	**6.48%**	13.95%

Table 3.24: Additional % of bit rate gains/lossess due to the use of the *PETW Bpp* estimator for the G and A quality ranges.

Images	Good		All	
	% Add.	% Tot.	% Add.	% Tot.
Lena	-0.95%	7.74%	0.98%	12.46%
Barbara	0.37%	18.42%	0.68%	15.81%
Goldhill	0.70%	12.70%	1.39%	11.65%
Boat	0.60%	6.90%	0.60%	7.49%
Mandrill	-0.02%	21.26%	-0.04%	16.63%
Balloon	-0.55%	8.79%	-0.31%	9.02%
Horse	0.25%	16.33%	0.44%	13.91%
Zelda	-0.23%	4.97%	2.40%	13.36%
Cafe	-1.00%	11.53%	-0.76%	10.44%
Bike	-0.58%	13.29%	-0.04%	12.46%
Woman	0.76%	11.36%	1.62%	10.41%
Deer	8.61%	40.79%	11.84%	36.55%
Big Tree	0.97%	9.98%	4.73%	12.68%

In Tables 3.23 to 3.24, we show in the column labeled as *%Add.* the percentage of additional gain or loss that could be obtained using the estimator in the PETW, and in the column labeled as *%Tot.* the percentage of bit rate saving that is obtained with respect to the rate obtained if we use the S-LTW encoder. Table 3.23 shows the values corresponding to the *Visually Lossless* and *Excellent* quality ranges, whereas Table 3.24 shows the values corresponding to the *Good* and *All* quality ranges.

So, the use of the *PETW Bpp* estimator in the PETW is able to produce additional bit rate savings in most of the images, increasing therefore the gain with respect to other encoders for those images. For some images instead there is a small loss of bit rate with respect to the equivalent xi value. Additionally, as the estimator uses the same algorithms used in the rate distortion stage of the S-LTW, no further computational cost or complexity is added to obtain the $E\xi$ value.

3.4.4 Performance results with dead zone estimation

In this section we will compare the PETW performance with the inclusion of the *PETW Bpp* estimator with other standard image encoders.

PETW has been compared with Kakadu 5.2.5 and SPIHT (Sphit 8.01) encoders with images in Table 3.21 with resolutions of $512x512$, and $2048x2560$ (higher resolutions failed in the SPIHT encoder used version).

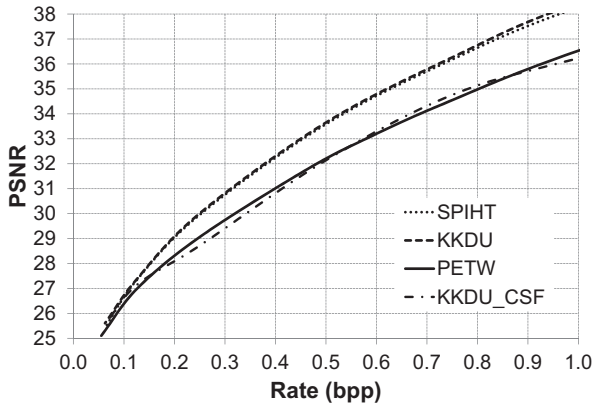


Figure 3.68: PSNR R/D comparison of Woman image encoded with PETW, SPIHT and Kakadu. Rates are in bpp.

When comparing with the Kadadu encoder, we perform two comparisons, one labeled as Kakadu_csf, which has enabled its perceptual weighting mode (with the perceptual weights presented in [242]), and the other one, labeled as Kakadu, without the perceptual weights.

Figure 3.68 shows the R/D comparison of the Woman image compressed with the PETW encoder, SPIHT, Kakadu and Kadadu_csf, using PSNR as the distortion metric. A misleading conclusion after looking at the R/D curves for the PETW, and Kakadu_csf, is that the encoding strategy used in these proposals are inappropriate, since their quality results are always lower than the ones for other encoders, specially at high bit rates. This is a consequence of using PSNR as distortion metric and not a QAM, when comparing perceptual enhanced encoders with non perceptual enhanced ones.

As an example of why measuring the quality of perceptual enhanced images with PSNR is misleading, we can see in Figure 3.69, a subjective comparison of the three encoders with a cropped region of the Woman test image compressed at 0.25 bpp. In this case the third image, encoded with PETW seems to have better subjective quality than the other two. This observation contradicts the conclusion obtained from Figure 3.68 that suggest that at 0.25 bpp PETW is worse than SPIHT and Kakadu. The same behavior can be observed as well with the other test images. So it is better not to trust in how PNSR assess quality and use instead a perceptual inspired quality assessment metric like VIF that, as stated in [210, 306] and in our tests, it has a better correlation with the human perception of quality.

Figures 3.70 and 3.71, show some of the VIF R/D results (R/D plots where VIF is the distortion metric) for some test images. As shown, PETW encoder

(a) SPIHT PSNR=29.95 dB

(b) Kakadu PSNR=30.01 dB

(c) PETW PSNR=29.11 dB

Figure 3.69: Subjective comparison of the Woman image encoded at 0.25 bpp with a) SPIHT, b) Kakadu, and c) PETW.

can achieve higher compression rates while maintaining the same perceptual quality than the other encoders, i.e., a bit rate saving is obtained at a desired quality when the PETW is used instead Kakadu or SPIHT.

Tables 3.25 to 3.27 show the rate savings obtained with PETW versus Kakadu, SPIHT and Kakadu_csf. These tables group the results also by image resolution. Results are expressed as percentage of saved rate in the aforementioned VIF intervals, i.e., Visually Lossless, Excellent, Good and All the range.

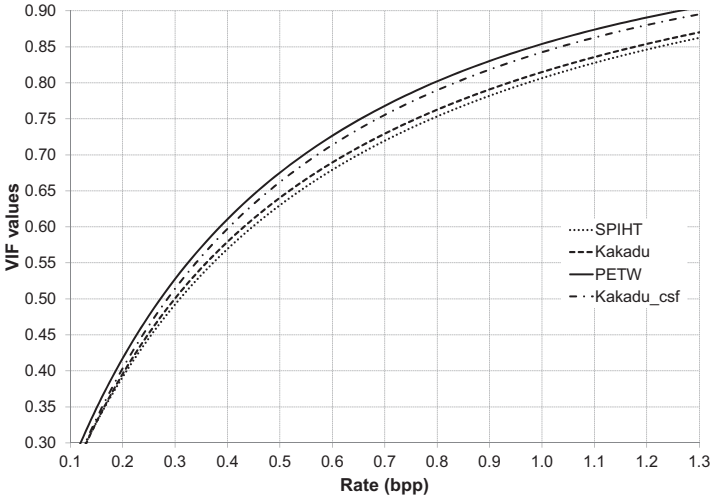
If we focus in the *All* quality range we see that the highest bit rate savings are obtained in comparison with the SPIHT encoder for both studied

Table 3.25: Rate savings of PETW versus Kakadu without perceptual weights

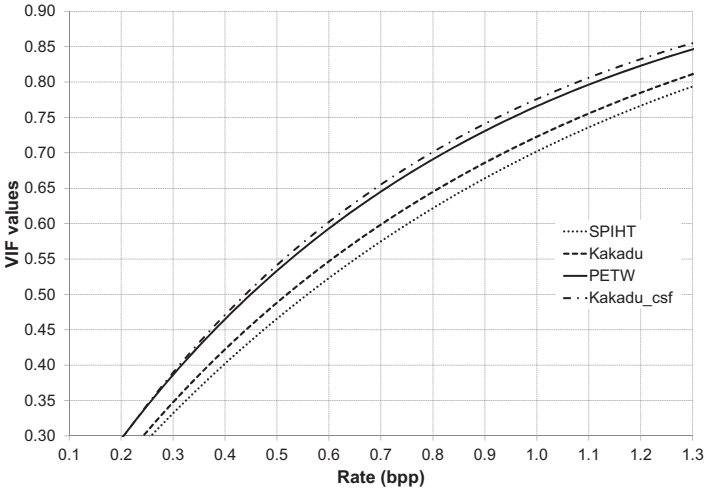
	PETW vs. KKDU		
	Excellent	Good	All
		512x512	
Lena	15.75%	9.98%	13.88%
Barbara	13.61%	15.91%	14.42%
Goldhill	6.93%	10.48%	8.10%
Boat	6.49%	7.53%	6.84%
Mandrill	21.40%	**27.06%**	23.45%
Balloon	10.35%	9.65%	10.11%
Horse	19.46%	19.11%	19.34%
Zelda	16.40%	10.25%	14.49%
Mean 512x512	**13.80%**	**13.75%**	**13.83%**
		2048x2560	
Cafe	10.37%	11.67%	10.84%
Bike	9.97%	12.06%	10.69%
Woman	5.16%	5.11%	5.14%
Mean 2048x2560	**8.50%**	**9.61%**	**8.89%**

Table 3.26: Rate savings of PETW versus SPIHT

	PETW vs. SPIHT		
	Excellent	Good	All
		512x512	
Lena	20.22%	12.90%	17.84%
Barbara	20.42%	24.49%	21.85%
Goldhill	16.73%	16.85%	16.77%
Boat	11.75%	13.58%	12.36%
Mandrill	23.26%	**26.38%**	24.39%
Balloon	11.06%	10.72%	10.94%
Horse	18.56%	20.27%	19.15%
Zelda	21.48%	7.74%	17.22%
Mean 512x512	**17.94%**	**16.62%**	**17.57%**
		2048x2560	
Cafe	13.99%	16.15%	14.77%
Bike	18.08%	20.92%	19.07%
Woman	11.98%	13.44%	12.50%
Mean 2048x2560	**14.69%**	**16.84%**	**15.45%**

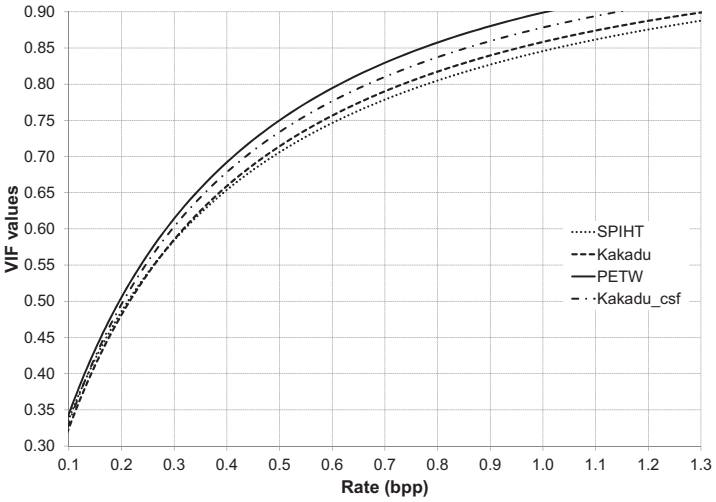


(a) Lena

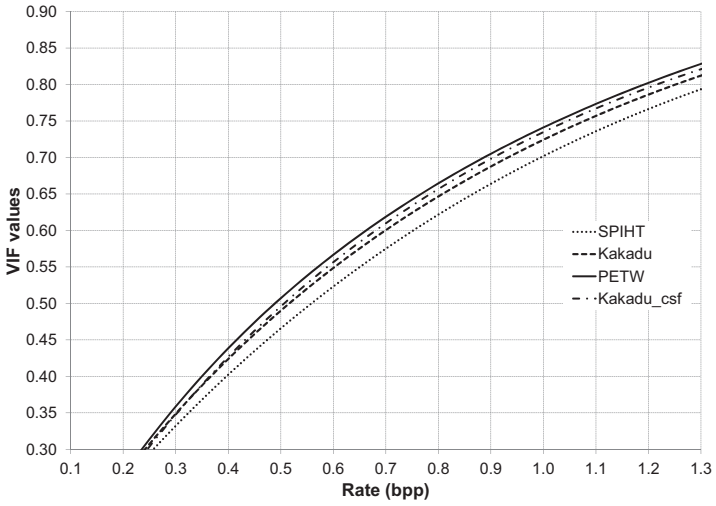


(b) Barbara

Figure 3.70: VIF R/D comparisons for the Lena and Barbara images.



(a) Zelda



(b) Woman

Figure 3.71: VIF R/D comparisons for the Zelda and Woman images.

Table 3.27: Rate savings of PETW versus Kakadu with perceptual weights

	PETW vs. KKDU_CSF		
	Excellent	Good	All
		512x512	
Lena	4.87%	5.30%	5.01%
Barbara	-2.98%	-1.86%	-2.59%
Goldhill	3.19%	1.91%	2.77%
Boat	-0.03%	2.07%	0.67%
Mandrill	1.35%	3.91%	2.28%
Balloon	0.73%	6.18%	2.57%
Horse	2.05%	6.37%	3.53%
Zelda	**6.88%**	4.41%	6.11%
Mean 512x512	**2.01%**	**3.54%**	**2.54%**
		2048x2560	
Cafe	0.22%	1.08%	0.53%
Bike	-1.12%	-2.14%	-1.47%
Woman	2.12%	3.70%	2.68%
Mean 2048x2560	**0.41%**	**0.88%**	**0.58%**

resolutions. The mean bit rate savings for the 512x512 resolutions is up to 15.57% and 15.54% for the 2048x2560 resolution. The maximum saving when comparing with SPIHT is obtained for the Mandrill image in the *Good* quality range with 26.38%.

When comparing with KKDU without perceptual weighting, the bit rate savings are also significant for the *All* quality range, in average 13.83% for small images and 8.89% for big ones. The maximum bit rate savings in comparison with this encoder are obtained also for the Mandrill image, and in the Good quality range with 27.06%

If the perceptual weightings are enabled in KKDU, for some images as Barbara or Bike, KKDU is performing better than our proposal, but in average for all the images there is still a bit rate saving of 2.54% for small images in the *All* range and 0.58% for big ones. The highest gains are obtained in the Excellent quality range for Zelda image with 6.88%.

But, in order to reduce the overall encoding time with respect to the S-LTW, we have also coded the PETW using a parallel implementation of the 2D-DWT transform on a GPU.

In [307], the authors analyzed the behavior of several parallel algorithms developed to compute the two-dimensional discrete wavelet transform using both, **OpenMP!** (**OpenMP!**) over a multicore platform, and **CUDA!** (**CUDA!**) over a GPU. So, we have used this implementation with the inclusion of the proposed PWM and the *PETW Bpp* estimator in [308], naming that encoder

Table 3.28: Speedup comparison by target bit rate and image size

PE_LTW Speed-up Comparisson		
Rates(bpp)	vs. SPIHT	vs. KKDU
Average for 512x512 image size		
1	2.76	0.42
0.5	3.74	0.61
0.25	4.80	0.66
0.125	6.86	0.83
Average for 2048x2560 image size		
1	1.73	0.33
0.5	5.22	0.38
0.25	4.31	0.44
0.125	4.39	0.53

version *Perceptually Enhanced LTW* encoder (PE_LTW).

As the 2D-DWT transform runs on a GPU, the overall encoding time is highly reduced compared to the sequential version of the same encoder (the PETW), obtaining maximum speed-ups of 6.86 for 512x512 images and 4.39 for 2048x2560 images. Comparing with SPIHT and Kakadu, the new proposal is clearly faster than SPIHT but needs additional optimizations to outperform Kakadu times. For details about how the wavelet transform has been parallelized, detailed encoding times, and detailed speedups per image, see [307, 308].

Table 3.28 compares the averaged speed ups for each image size in the test set at different compression rates. The PE_LTW is faster than SPIHT regardless of the target rate, and for any image size. However the Kakadu encoder is still faster than the PE_LTW. The reason is that, although the PE_LTW runs its DWT stage over the GPU, it is the only optimized stage in the whole encoder. By contrast, all encoding stages in the Kakadu 5.2.5 are fully optimized. Besides of the use of multi-thread and multi core hardware capabilities, Kakadu uses also processor intrinsic capabilities like MMX/SSE/SSE2/SIMD and uses a very fast multicomponent transform, i.e., block transform, which is well suited for parallelization.

Chapter 4

Conclusions and future work
Conclusiones y trabajo futuro

Contents

4.1 Conclusions

In this section, we conclude this thesis and summarize some of the main contributions introduced in this thesis.

In Chapter 2, we proceeded with the state-of-the-art in the field of QAM, analyzed the most important aspects of the human visual system that researchers have included in their metrics design, and performed a classification of the metrics into frameworks attending to the way and methods that the metrics use in order to emulate how the human visual system assesses quality.

We also review how to compare QAM, and discussed some issues that must be taken into account in those comparisons. The metrics under comparison were DMOSp-PSNR, MSSIM, VIF, NRJPEG2000, RRIQA, NRJPEGQS, and VQM.

In our correlation analysis, the metric that obtained higher correlation with DMOS was the VIF metric. In order to be able to fairly compare the QAM, they must be first moved to a common scale. We used the DMOSp scale, which is a prediction of the real DMOS. To move a metric into the DMOSp scale, a parametric equation must be used. We have published the parameters obtained in our study here so that the results can be replicated.

Then we performed a comparison of the behavior of the QAM under study in two environments, specifically image and video compression, and mobile networks with packet losses.

- For the compression environment, we analyzed the performance of H.264/AVC [224], Motion-JPEG2000 [225], and Motion-LTW [226] in intra mode.

  We made the comparisons using R/D curves changing the PSNR with each of the evaluated metrics using the different encoders. We conclude that the VIF metric ranks the performance of the tested encoders more accurately, i.e., orders the encoders by quality in the same order that the subjective ranking does for any compression level.

  If we want to replace the PSNR metric with one of the evaluated QAMs in order to test the performance of a new encoder design, then if the reference (image or video) is available and accuracy is needed, the choice is the VIF metric followed by MSSIM. If computational time is critical, then the choice is VQM and MSSIM. If the reference is not available, the choice is RRIQA.

  We use the VIF metric in the comparisons made in Chapter 3 as we work in a compression environment, with access to the reference images or

sequences, i.e., in full reference mode, and the accuracy is more important than obtaining the perceptual quality value faster.

- In the loss-prone environment, we analyzed the metrics behavior when measuring reconstructed video sequences encoded and delivered through error prone wireless networks, like MANETs.

  – NR metrics are not able to properly detect and measure the sharp quality drop due to the loss of several consecutive frames.

  – The RR metric has a non-deterministic behavior in the presence of packet losses, having difficulties to identify and measure this effect at moderate to high compression rates.

  – MSSIM, DMOSp-PSNR, and VIF exhibit similar behavior in all cases.

  So, for the loss-prone environment, we propose the use of the MSSIM metric as a trade-off between accuracy and computational cost.

In Chapter 3, we presented a comprehensive study of the perceptual coding techniques. The most widely used technique is the use of the CSF with several approximations found.

One of the motivations of this work was to avoid the need to perform continuous subjective tests. This motivation led us to choose some approximations that use the CSF without the need to perform subjective tests in order to determine the perceptual importance of the wavelet coefficients via a perceptual weighting matrix. These values are extracted directly from a CSF model.

Following the methods in those reference works, we increased the granularity of the reference work into a subband decomposition level. So, a subband weighting matrix is proposed, whose weights are obtained in an alternative way that optimizes the perceptual R/D behavior, i.e., using a QAM as distortion metric. We performed a comprehensive study of different ways to obtain the perceptual quantization matrix for at-threshold compression. We also proposed a normalization strategy for that matrix in order to obtain a perceptual subband weighting matrix.

The PWM-S4 weighting matrix, which is the best performing one among our proposals, was implemented into the S-LTW and the results have been compared with the reference matrices, one for a level decomposition and another for a subband decomposition.

Compared with the level decomposition of the reference, our proposal obtains bit rate savings on average for the test image set of 7.22% at the

Visually Lossless threshold, 6.50% in the *Excellent* quality range, 4.15% in the *Good* range, and 5.69% in the *All* quality range. Otherwise, the best results are 11.50%, 10.08%, and 7.64% for *Visually Lossless*, *Excellent*, and *Good* ranges, respectively.

Compared with the subband decomposition of the reference, our proposal obtains bit rate savings on average for the test image set of 0.96% at the *Visually Lossless* threshold, 3.40% in the *Excellent* quality range, 8.14% in the *Good* range, and 5.00% in the *All* quality range. Otherwise, the best results are 8.73%, 9.73%, and 13.41% for *Visually Lossless*, *Excellent*, and *Good* ranges, respectively.

We have finally presented the PETW encoder, which is an evolution of the S-LTW encoder into a perceptual image wavelet encoder that reveals the importance of exploiting the contrast sensitivity function by means of an accurate perceptual weighting of the wavelet coefficients.

In a first version of the PETW encoder, only the PWM-S4 was included and the quantization strategy of the encoder has been changed. We implemented a uniform variable dead zone quantizer (UVDZQ) into the PETW, changing the original two-stage quantizer. We prove the equivalence of the PSNR R/D performance with the original S-LTW encoder, and then we compare the new proposal with the M-LTW original video encoder and other well-known video encoders running in intra using a *motion* version of the PETW that we called M-PETW.

The best results of the comparison with the M-LTW show bit rate savings for:

- The QCIF resolution of 10.63%, 10.10%, 10.35%, and 10.19% for the *Visually Lossless*, *Excellent*, *Good*, and *All* quality ranges, respectively.
- The CIF resolution of 10.14%, 12.14%, 13.59%, and 11.63% for the *Visually Lossless*, *Excellent*, *Good*, and *All* quality ranges, respectively
- The ITU-D1 resolution of 10.05%, 8.34%, 4.39%, and 7.99% for the *Visually Lossless*, *Excellent*, *Good*, and *All* quality ranges, respectively
- The HD resolution of 16.22%, 14.05%, 5.54%, and 12.45% for the *Visually Lossless*, *Excellent*, *Good*, and *All* quality ranges, respectively

Regarding the comparison of M-PETW versus the rest of the encoders, the maximum bit rate average saving is produced at the *Visually Lossless* threshold for the HD frame size, except when compared with X.264. With that encoder, the best gain is obtained at the *Good* quality level. The best averaged bit rate

savings are 10.16%, 22.09%, 11.40%, 23.11%, and 10.69% in the comparisons with M-JASPER, M-SPIHT, KKDU, X.264, and H.264, respectively. But we obtain a maximum up to 28.01% at the *Good* quality level when compared with X.264, 22.09% in the *Visually Lossless* threshold when compared with M-SPIHT or even up to 12.85% when compared with H.264.

As shown in these comparisons, the proposed perceptual weighting matrix and implemented in the PETW encoder, obtains higher bit rate savings on average as the frame resolution increases.

In the final version of the PETW encoder, we implemented a new proposal of an image adaptive dead zone size estimator. Results confirm the importance of using an optimum dead zone size for each image to obtain a better quality of the reconstructed image.

The image adaptive dead zone size estimator is developed in order to obtain the best R/D performance when the distortion metric is the VIF metric. The methods used in this proposal can be extrapolated for use any other distortion metric instead. Several estimators were tested and the best performing one is PETW encoder dependent. One of the proposed dead zone size estimators is, however encoder independent, so with some adaptations it could be used in other wavelet and DCT-based encoders.

The use of the image adaptive dead zone size estimator in the PETW produces additional bit rate savings, and, depending on the image, up to 16.11%, 13.34%, 8.61%, and 11.84% in the *Visually Lossless*, *Excellent*, *Good*, and *All* quality ranges, respectively.

The PETW is very competitive in terms of perceptual quality, measured with the VIF QAM, being able to obtain important bit rate savings regardless of the image resolution, and at any bit rate, when compared with S-LTW, SPIHT, and Kakadu (with and without its perceptual weighting mode enabled). The PETW encoder is able to produce a quality equivalent image with respect to the other encoders with a reduced rate.

When compared with other encoders, the average bit rate savings are:

- With SPIHT:
  - For the 512x512 resolution, they are 17.94%, 16.62%, and 17.57% for the *Excellent*, *Good*, and *All* quality ranges, respectively, with a maximum savings of 26.38% in the *Good* range.
  - For the 2048x2560 resolution, they are 14.69%, 16.84%, and 15.45% for the *Excellent*, *Good*, and *All* quality ranges, respectively, with a maximum savings of 20.92% in the *Good* range.

- With Kakadu without perceptual weighting:
  - For the 512x512 resolution, they are 13.80%, 13.75%, and 13.83% for the *Excellent*, *Good*, and *All* quality ranges, respectively, with a maximum savings of 27.06% in the *Good* range.
  - For the 2048x2560 resolution, they are 8.50%, 9.61%, and 8.89% for the *Excellent*, *Good*, and *All* quality ranges, respectively, with a maximum savings of 12.06% in the *Good* range.
- With Kakadu with perceptual weighting:
  - For the 512x512 resolution, they are 2.01%, 3.54%, and 2.54% for the *Excellent*, *Good*, and *All* quality ranges, respectively, with a maximum savings of 6.58% in the *Good* range.
  - For the 2048x2560 resolution, they are 0.41%, 0.88%, and 0.58% for the *Excellent*, *Good*, and *All* quality ranges, respectively, with a maximum savings of 3.70% in the *Good* range.

The PETW encoder does not increases the overall encoding time with respect to the S-LTW, the dead zone size estimator uses an estimation of the bits per pixel needed that the already implemented rate control algorithms of the S-LTW provides, and the perceptual weighting is a simple multiplication of the wavelet coefficients with the corresponding perceptual scaling factor.

As a final conclusion, we have to remark that we have covered all the proposed objectives for this thesis, although some experiments and future work could be done as we expose next.

4.2 Conclusiones

En esta sección vamos a resumir las contribuciones más relevantes de esta tésis.

En el Capítulo 2, realizamos un estudio del arte sobre las métricas de valoración subjetiva de la calidad (Quality Assessment Metrics - QAM), analizando los aspectos más importantes del sistema visual humano (HVS) que los investigadores en este campo han incluido en el diseño de sus métricas, y realizamos una clasificación de las métricas agrupÃ¡ndolas atendiendo a la forma y métodos que las métricas utilizan para emular la forma en la que el sistema visual humano realizar la valoración de la calidad.

También revisamos cómo se deben comparar las QAM, y discutimos ciertas cuestiones que tienen que ser tomadas en cuenta en estas

comparaciones. Las metricas comparadas fueron: DMOSp-PSNR, MSSIM, VIF, NRJPEG2000, RRIQA, NRJPEGQS y VQM.

En nuestros análisis de correlación, la métrica que obtuvo mayor correlación con el DMOS fue la VIF. Para realizar una comparación justa entre métricas, éstas deben trasladarse primero a una escala común. Nosotros hemos utilizado la escala DMOSp, que es una predicción de los valores reales DMOS. Para trasladar una métrica a la escala DMOSp, se necesita una ecuación parametrica. Nosotros hemos puclicado aquí los parÃ¡metros obtenidos en nuestro estudio de forma que los resultados puedan ser reproducidos.

Después realizamos una comparación del comportamiento de las métricas (QAM) en dos entornos, concretamente compresión de imagen y video y redes móviles con pérdida de paquetes.

- En el entorno de compresión analizamos el rendimiento del H.264/AVC [224], Motion-JPEG2000 [225] y Motion-LTW [226] en modo intra.

  Realizamos las comparaciones utilizando curvas R/D cambiando el PSNR por cada una de las métricas evaluadas y utilizando diferentes codificadores. Concluimos que la métrica VIF ordena por calidad el rendimiento de los codificadores probados de manera mÃ¡s precisa, es decir, ordena los codificadores por calidad en el mismo orden que una ordenación subjetiva para cualquier nivel de compresión.

  Si pretendemos reemplazar la métrica PSNR por una de las métricas QAM analizadas para valorar el rendimiento del diseño de un nuevo codificador, entonces, si la referencia (imagen o video) estÃ¡ disponible y es necesaria precisión la elección es la métrica VIF seguida por la MSSIM. Si el tiempo de cÃ¡lculo es crítico, entonces la elección es VQM y MSSIM. Si la referencia no estÃ¡ disponible, entonces la elección es RRIQA.

  Usaremos la métrica VIF en las comparaciones realizadas en el Capítulo 3 puesto que trabajamos en un entorno de compresión, con acceso a la imagen o sequencia de referencia, es decir en modo *full reference*, y estamos diseñando una propuesta de codificación por lo que la precisión es mÃ¡s importante que obtener el valor perceptual de calidad mÃ¡s rÃ¡pido.

- En el entorno propenso a pérdidas, analizamos el comportamiento de las métricas al medir la calidad de secuencias de video reconstruidas enviadas a traves de redes móviles propensas a la pérdida de paquetes como las MANETs.

  – Las métricas NR no son capaces de detectar apropiadamente y medir la

abrupta caida de calidad producida por la pérdida de varios frames consecutivos.

– La métrica RR tiene un comportamiento no determinista ante la pérdida de paquetes, mostrando dificultades para identificar y medir este efecto para ratios de compresión de moderados a altos.
– Las métricas MSSIM, DMOSp-PSNR y VIF muestran un comportamiento similar en todos los casos.

Por tanto para el entoro propenso a pérdidas proponemos el uso de la métrica MSSIM como compromiso entre precisión y coste computacional.

En el Capítulo 3 presentamos un completo estudio de las técnicas de codificación perceptual. La técnica mÃ¡s ampliamente utilizada es utilizar la CSF (Contrast Sensitivity Function) para lo cual encontramos varias aproximaciones.

Una de las motivaciones de este trabajo fue tratar de evitar la necesidad de realicar continuos test subjetvios. Esta motivación nos llevó a elegir aproximaciones qeu usan la CSF sin necesidad de realizar tests subjetivos para determinar la importancia perceptual de los coeficientes wavelet, utilizando una matriz perceptual de pesos. Estos valores se obtienen directamente de un modelo de la CSF.

Siguiendo los métodos del trabajo de referncia hemos incrementado la granularidad de éste a el nivel de descomposicion en subbandas. Por lo que proponemos una matriz de pesos por subbandas cuyos pesos se obtienen de una forma alternativa que optimiza el comportamiento R/D perceptual, es decir usando una métrica QAM como métrica de distorsión. Realizamos un completo estudio de diferentes formas de obtención de la matriz de quantización para una compresión at-threshold. Proponemos también una estrategia de normalización para esta matriz para convertirla en una matriz perceptual de pesos por subbandas.

La matriz PWM-S4, que es la que mejor rendimiento ha demostrado de entre nuestras propuestas, se ha implementado en el codificador S-LTW y los resultados han sido comparados con las matrices de referencia, una para una descomposición por niveles y otra por subbandas.

En la comparación con la matriz de referencia por niveles, nuestra propuesta obtiene unos ahorros en rate, que en promedio para el conjunto de imÃ¡genes de prueba es de un 7.22% en el umbral *Visually Lossless*, un 6.50% en el rango de calidad *Excellent*, un 4.15% en el rango *Good* y un 5.69% en el rango de calidad *All*. Los mejores resultados han sido sin embargo de un

11.50%, un 10.08% y un 7.64% para lor rangos *Visually Lossless*, *Excellent* y *Good* respectivamente.

En la comparación con la matriz de referencia por subbandas, nuestra propuesta obtiene unos ahorros en rate, que en promedio para el conjunto de imÃ¡genes de prueba es de un 0.96% en el umbral *Visually Lossless*, un 3.40% en el rango de calidad *Excellent*, un 8.14% en el rango *Good* y un 5.00% en el rango de calidad *All*. Los mejores resultados han sido sin embargo de un 8.73%, un 9.73% y un 13.41% para los rangos *Visually Lossless*, *Excellent* y *Good* respectivamente.

Finalmente hemos presentado el codificador PETW. Una evolución del codificador S-LTW en un codificador perceptual de imagen, basado en wavelets, que manifiesta la importancia de explotar la funcion de sensibilidad al contraste por medio de una ponderación perceptual muy precisa de los coeficientes wavelet.

En una primera vesión del codificador PETW, sólo se ha incluido la matriz PWM-S4 y se ha cambiado la estratégia de quantización del codificador. Hemos implementado un quantizador uniforme con dead zone variable (UVDZQ) en el PETW, sustituyendo el cuantizador en dos fases original. Hemos demostrado la equivalencia del rendimiento R/D en PSNR con el codificador S-LTW original y entonces hemos comparado la nueva propuesta contra el codificador de video M-LTW original y contra otros codificadores de video muy populares corriendo en modo intra, para lo que hemos usado una versión *motion* del codificador PETW y que llamamos M-PETW.

Los mejores resultados de las comparaciones con el M-LTW muestran ahorros de rate para:

- La resolución QCIF en un 10.63%, un 10.10%, un 10.35%, y un 10.19% para los rangos de calidad *Visually Lossless*, *Excellent*, *Good*, y *All* respectívamente.

- La resolución CIF en un 10.14%, un 12.14%, un 13.59%, y un 11.63% para los rangos de calidad *Visually Lossless*, *Excellent*, *Good* y *All* respectívamente.

- La resolución ITU-D1 en un 10.05%, un 8.34%, un 4.39%, y un 7.99% para los rangos de calidad *Visually Lossless*, *Excellent*, *Good* y *All* respectívamente.

- La resolución HD en un 16.22%, un 14.05%, un 5.54%, y un 12.45% para los rangos de calidad *Visually Lossless*, *Excellent*, *Good* y *All* respectívamente.

En cuanto a la comparación del M-PETW versus el resto de codificadores, los mÃ¡ximos ahorros de rate se producen en el umbral *Visually Lossless* para la resolución HD, excepto cuando comparamos con el X.264. Con este codificador las mejores ganancias se obtienen en el rango de calidad *Good*. Los mejores ahorros de rate en promedio que se obtienen son de un 10.16%, un 22.09%, un 11.40%, un 23.11%, y un 10.69% en las comparaciones con M-JASPER, M-SPIHT, KKDU, X.264, y con H.264 respectívamente. Pero se obtienen unos mÃ¡ximos de hasta un 28.01% en el nivel de calidad *Good* cuando comparamos con X.264, un 22.09% en el umbral *Visually Lossless* cuando comparamos con M-SPIHT o incluso un 12.85% cuando comparamos con H.264.

Como se ve en estas comparaciones, la matriz perceptual de pesos propuesta e implementada en el codificador PETW obtiene mayores ahorros en rate en promedio a medida que la resolución crece.

En la versión final del codificador PETW implementamos una nueva propuesta de un estimador adaptativo del ancho del *dead zone*. Los resultados confirman la importancia de utilizar un ancho óptimo del *dead zone*para cada imagen de forma que se obtenga una mayor calidad en la imagen reconstruida.

El estimador adaptativo por imagen del ancho del *dead zone* se ha desarrollado para obtener el mejor rendimiento R/D cuando la metrica de distorsión utilizada es la VIF. Los métodos utilizados en esta propuesta pueden extrapolarse sin embargo, para ser usados con cualquier otra métrica de calidad. Se han probado diferentes estimadores y el que mejor rendimiento ofrece es dependiente del codificador PETW. Sin embargo, una de las propuesta de estimadores del ancho del *dead zone* es independiente del codificador, por lo que con las adaptaciones oportunas podría ser usado en otros codificadores basados en wavelet o incluso en DCT.

El uso del estimador adaptativo en el PETW proporciona ahorros de rate adicionales, que dependiendo de la imagen llegan hasta un 16.11%, un 13.34%, un 8.61%, y un 11.84% en los rangos de calidad *Visually Lossless*, *Excellent*, *Good* y *All* respectívamente.

El PETW es muy competitivo en terminos de calidad perceptual, medida con la métrica VIF, siendo capaz de obtener importantes ahorros de rate independientemente de la resolución de la imagen y a cualquier tasa de bits, cuando es comparado con S-LTW, SPIHT, y Kakadu (con y sin su modo de ponderación perceptual activo). El codificador PETW es capaz de producir una imagen con una calidad perceptual equivalente con respecto a otros codificadores reduciendo la tasa de bits.

En las comparaciones con otros codificadores los ahorros de rate son:

- Con SPIHT:
  - Para la resolución 512x512 un 17.94%, un 16.62% y un 17.57% para los rangos de calidad *Excellent, Good* y *All* respectívamente y con un mÃ¡ximo de ahorro de un 26.38% en el rango *Good*.
  - Para la resolución 2048x2560 un 14.69%, un 16.84% y un 15.45% para los rangos de calidad *Excellent, Good* y *All* respectívamente y con un mÃ¡ximo de ahorro de un 20.92% en el rango *Good*.
- Con Kakadu sin ponderación perceptual:
  - Para la resolución 512x512 un 13.80%, un 13.75% y un 13.83% para los rangos de calidad *Excellent, Good* y *All* respectívamente y con un mÃ¡ximo de ahorro de un 27.06% en el rango *Good*.
  - Para la resolución 2048x2560 un 8.50%, un 9.61% y un 8.89% para los rangos de calidad *Excellent, Good* y *All* respectívamente y con un mÃ¡ximo de ahorro de un 12.06% en el rango *Good*.
- Con Kakadu con ponderación perceptual:
  - Para la resolución 512x512 un 2.01%, un 3.54% y un 2.54% para los rangos de calidad *Excellent, Good* y *All* respectívamente y un mÃ¡ximo de ahorro de un 6.58% en el rango *Good*.
  - Para la resolución 2048x2560 un 0.41% un 0.88% y un 0.58% para los rangos de calidad *Excellent, Good* y *All* respectivamente y un mÃ¡ximo de ahorro de un 3.70% en el rango *Good*.

El codificador PETW no retrarda el tiempo de codificación respecto al S-LTW porque el estimador adaptativo del ancho del dead zone utiliza los algoritmos de rate control ya implementados en el S-LTW y la ponderación perceptual es una simple multiplicación de los coeficientes wavelet por su factor de escala correspondiente.

Como última conclusión, remarcar que se han realizado todos los objetivos propuestos para esta tesis, aunque algunos experimentos y algún trabajo futuro se puede realizar como se expone a continuación.

4.3 Future work

There are some tasks and more research related with the subject of this thesis that could be done as future work, such as:

- Extend the study of the QAM with new metrics published recently.
- Analyze comprehensively the behavior and performance of the new metrics, and the ones studied here, when facing ultra high definition resolutions.
- Include chrominance models of the CSF in the development of the perceptual weighting matrix.
- Include a luminance and masking model in the PETW. Analyze the behavior in the PETW of different proposals, optimize them to be used in the PETW, and try to make them image adaptive.
- Improve the adaptive dead zone size estimator trying also to make it encoder independent.
- Analyze how to include the methods followed to obtain the perceptual weighting matrix into the HEVC encoder.

Appendix I

Acronyms

ACM	Adaptive Coefficient Modification
ACR	Absolute Category Rating
AMD	Adaptive Modification Distortion function
AMP	Asymmetric Motion Partitions
ANSI	American National Standards Institute
APIC	Adaptive Picture Image Coding
ATM	Asynchronous Transfer Mode
AVC	Advanced Video Coding
BD-PSNR	Bjontegaard Delta PSNR
bpp	bits per pixel
CAVLC	Context Based Adaptive Variable Length Coding
Cb	Chrominance blue
CCIR	International Radio Consultative Committee - Comité Consultatif International des Radiocommunications
CIE	International Commission on Illumination
cpd	Cycles per Degree
CPU	Central Processing Unit
Cr	Chrominance red
CSF	Contrast Sensitivity Function
CU	Coding Unit
CW-SSIM	Complex Wavelet SSIM
dB	decibel
DCQ	Dynamic Contrast-Based Quantization
DCR	Degradation Category Rating
DCT	Discrete Cosine Transform
DMOS	Difference Mean Opinion Score

DMOSp	Predicted DMOS
DSCQS	Double-Stimulus Continuous Quality-Scale
DSIS	Double-Stimulus Impairment Scale
DVQ	Digital Video Qualtiy
DVR	Digital Video Recording
DWT	Discrete Wavelet Transform
DZ	Dead Zone
EBCOT	Embedded Block Coding with Optimized Truncation
ESF	Error Sensitivity Framework
EZW	Embedded Zero-tree Wavelet
FFT	Fast Fourier Transoform
FGS	Fine Grain Scalability
FIR	Finite Impulse Response
FMO	Flexible Macroblock Ordering
FR	Full Reference
FRExt	Fidelity Range Extension
FRTV-I	Full-Reference Television
FTP	File Transfer Protocol
FWQI	Foveated Wavelet Image Quality Index
GB	Giga Byte
GGD	Generalized Gaussian Density
GIS	Geographic Information System
GOP	Group of Pictures
GQMF	Generalized Quadrature Median Filter
GSM	Gaussian Scale Mixtures
HD	High Definition

HEVC	High Efficiency Video Coding
HIQM	Hybrid Image Qualitiy Metric
HMM	Hidden Markov Model
HRC	Hypothetical Reference Circuit
HVS	Human Visual System
I	Intra
IAF	Information Allocation Function
IEEE	Institute of Electrical and Electronics Engineers
IFC	Information Fidelity Criterion
IIR	Infinite Impulse Response
IPQ	Instrumental Picture Quality
ISF	Invariant Scaling Factor
ISO	International Organization for Standardization
ITS	Institute for Telecommunication Sciences
ITU-R	International Telecommunication Union Recommendation
JCT-VC	Joint Collaborative Team on Video Coding
JND	Just Noticeable Difference
JP2K	JPEG2000
JPEG	Joint Photographic Experts Group
Kb/s	Kilobits per second
KLD	Kullback-Leiber Distance
LCU	Larger Coding Unit
LIP	List of Insignificant Pixels
LIS	List of Insignificant Sets
LSB	Least Significant Bit
LSF	Line Spread Function

LSP	List of Significant Pixels
LTW	Lower Tree Wavelet
LZC	Layered Zero Coding
MANET	Mobile Ad Hoc Networks
Mbps	Mega bits per second
MC	Motion Compensation
MC-ed LS	Motion Compensated Lifting Schemes
MDIS	Mode dependent intrasmoothing
ME	Motion Estimation
M-LTW	Motion LTW
MM-I	VQEG Multimedia Phase I
MNDSS	Minimum Noticeable Quantizer Step Size
MOS	Mean Opinion Score
MPEG	Moving Picture Experts Group
M-PETW	Motion-PETW
MPQM	Moving Picture Quality Metric
MSB	Most Significant Bit
MSE	Mean Squared Error
MSSIM	Mean SSIM
M-SSIM	Multi-Scale Structural SIMilarity
MTF	Modulation Transfer Function
MV	Motion Vector
MVC	Multiview Video Coding
NR	No Reference
NRJPEG2000	No-Reference JPEG2000 Quality Assessment
NRJPEGQS	No-Reference JPEG Quality Score

NTIA	National Telecommunications and Information Administration
OBR	Objective Blocking Rating
OR	Outlier Ratio
P	Predicted
PC	Pair Comparison
PCC	Pearson Correlation Coefficient
PCRD	Post-Compression Rate Distortion
PDF	Portable Document Format
PDF	Probability Density Function
PDM	Perceptual Distortion Metric
PETW	Perceptually Enhanced Tree Wavelet
PIC	Perceptual Image Codec
PQM	Perceptual Quantization Matrix
PSF	Point Spread Function
PU	Prediction Unit
PWM	Perceptual Weighting Matrix
QM	Quantization Matrix
QMF	Quadrature Mirror Filter
Qstep	Quantization step
RBSP	Raw Byte Sequence Payload
RMSE	Root Mean Squared Error
ROI	Regions Of Interest
RR	Reduced Reference
RRIQA	RR Image Quality Assesment
RTP	Real-time Transport Protocol

SD	Standard Deviation
SDSCE	Simultaneous Double Stimulus for Continuous Evaluation
SFQ	Space-Frequency Quantization
SI	Switching I
SNR	Signal to Noise Ratio
SP	Switching P
SPIHT	Set Partitioning In Hierarchical Trees
SROCC	Spearman Rank Order Correlation Coefficient
SS	Single-Stimulus
SSCQE	Single Stimulus Continuous Quality Evaluation
SSD	Solid State Drives
SSE	Sum of Squares due to Error
SSIM	Structural SIMilarity
TCP	Transmission Control Protocol
TU	Tansform Unit
UQI	Universal Quality Index
USDZQ	Uniform Scalar Dead Zone Quantizer
USQ	Uniform Scalar Quantizer
UTCQ	Universal Trellis Coded Quantizer
UVDZQ	Uniform Variable Dead Zone Quantizer
UVDZQ	Uniform Variable Dead Zone Quantizer
VCEG	Video Coding Experts Group
VIF	Visual Information Fidelity
VLC	Variable Length Coding
VPSF	Visual Progressive Single Factor
VQEG	Video Quality Experts Group

VQM	Video Quality Measurements Techniques
VT	Visual Thresholds
WMSE	Weighted Mean Squared Error
WMSE	Weighted MSE
WPP	Wavefront Parallel Processing
WT	Wavelet Transform

Appendix II

Kodak images set

Img01 Img02 Img03

Img04 Img05 Img06

Img07 Img08 Img09

Img10 Img11 Img12

Figure II.1: Kodak image set (768x512)

Img13 Img14 Img15

Img16 Img17 Img18

Img19 Img20 Img21

Img22 Img23

Figure II.2: Kodak image set (768x512)

Appendix III

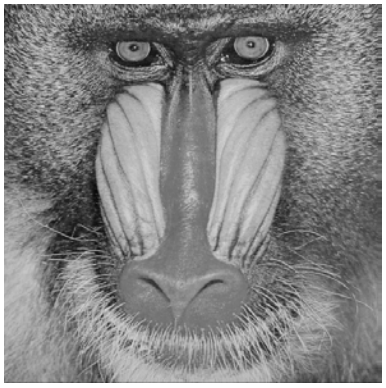
Test images

Barbara (512x512)

Lena (512x512)

Boat (512x512)

GoldHill (512x512)



Mandrill (512x512)

Horse (512x512)

Figure III.1: Test image set

Zelda(512x512)

Bike(2048x2560)

Cafe(2048x2560)

Deer (3968x2560)

Big Tree (6016x4480)

Big Building (7168x5376)

Figure III.2: Test image set

Appendix IV

Test Videos

Foreman (QCIF and CIF)

News (QCIF and CIF)

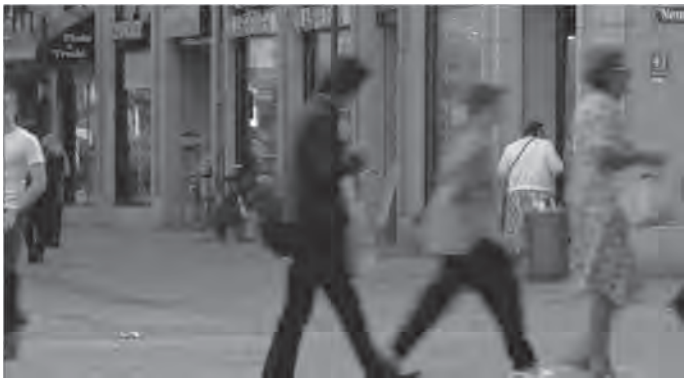
Container (QCIF and CIF)

Hall (QCIF and CIF)

Mobile (ITU)

Ducks Take Off (HD)

Station2 (HD)



Pedestrian Area (HD)

Figure IV.1: Test video sequences set

Appendix V

Articles

Según la normativa vigente de la Universidad Miguel Hernández en las siguientes páginas se recogen los artículos presentados junto con esta memoria en su formato original escalado.

According to the current regulations of the Miguel Hernández University the following pages include the articles that support the present work, which are included in a scaled version of its original format.

- **On the Performance of Video Quality Assessment Metrics under Different Compression and Packet Loss Scenarios**

  Miguel O. Martínez-Rach, Pablo Piñol, Otoniel M. López, Manuel PerezMalumbres, José Oliver, and Carlos Tavares Calafate Hindawi Publishing Corporation - The Scientific World Journal Volume 2014, Article ID 743604, 18 pages http://dx.doi.org/10.1155/2014/743604

  The Sicentific World Journal is indexed in the JCR with next data:

  Journal Ranking: For 2013, the journal Scientific World Journal has an **Impact Factor of 1.219**.
  Category Name: **MULTIDISCIPLINARY SCIENCES**
  Total Journals in Category: **55**
  Journal Rank in Category: **16**
  Quartile in Category: **Q2**

- **Enhancing LTW image encoder with perceptual coding and GPU-optimized 2D-DWT transform**

  Miguel O. Martínez-Rach, Otoniel López-Granado, Vicente Galiano, Hector Migallón, Jesús Llor and Manuel P. Malumbres EURASIP Journal on Advances in Signal Processing 2013, 2013:141 http://asp.eurasipjournals.com/content/2013/1/141

  The Journal on Advances in Signal Processing is indexed in the JCR with next data:

  Journal Ranking: For 2013, the journal EURASIP Journal on Advances in Signal Processing has an **Impact Factor of 0.808**.
  Category Name: **ENGINEERING, ELECTRICAL & ELECTRONIC**
  Total Journals in Category: **247**
  Journal Rank in Category: **164**
  Quartile in Category: **Q3**

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 743604, 18 pages
http://dx.doi.org/10.1155/2014/743604

Research Article

On the Performance of Video Quality Assessment Metrics under Different Compression and Packet Loss Scenarios

Miguel O. Martínez-Rach,[1] Pablo Piñol,[1] Otoniel M. López,[1] Manuel Perez Malumbres,[1] José Oliver,[2] and Carlos Tavares Calafate[2]

[1] *Department of Physics and Computer Engineering, the Miguel Hernández University, Avenida de Universidad s/n, Elche, 03202 Alicante, Spain*
[2] *Department of Computer Engineering, Polytechnic University of Valencia, Camino de Vera s/n, Building G1, 46022 Valencia, Spain*

Correspondence should be addressed to Miguel O. Martínez-Rach; mmrach@umh.es

Received 7 January 2014; Accepted 14 April 2014; Published 20 May 2014

Academic Editors: H. Su and W. Su

When comparing the performance of video coding approaches, evaluating different commercial video encoders, or measuring the perceived video quality in a wireless environment, Rate/distortion analysis is commonly used, where distortion is usually measured in terms of PSNR values. However, PSNR does not always capture the distortion perceived by a human being. As a consequence, significant efforts have focused on defining an objective video quality metric that is able to assess quality in the same way as a human does. We perform a study of some available objective quality assessment metrics in order to evaluate their behavior in two different scenarios. First, we deal with video sequences compressed by different encoders at different bitrates in order to properly measure the video quality degradation associated with the encoding system. In addition, we evaluate the behavior of the quality metrics when measuring video distortions produced by packet losses in mobile ad hoc network scenarios with variable degrees of network congestion and node mobility. Our purpose is to determine if the analyzed metrics can replace the PSNR while comparing, designing, and evaluating video codec proposals, and, in particular, under video delivery scenarios characterized by bursty and frequent packet losses, such as wireless multihop environments.

1. Introduction

In the past years, the development of novel video coding technologies has spurred the interest in developing digital video communications, where evaluation mechanisms to assess the video quality play a major role in the overall design of video communication systems.

The most reliable way of assessing zthe quality of a video is subjective evaluation, because human beings are the ultimate receivers in most applications. The mean opinion score (MOS), which is a subjective quality metric obtained from a number of human observers, has been regarded for many years as the most reliable form of quality measurement. However, the MOS method is too cumbersome, slow, and expensive for most applications. Objective quality assessment metrics (QAM) are valuable because they provide video designers and standard organizations with means for making meaningful quality evaluations without convening viewer panels.

Recently, new objective image and video quality metrics have been proposed. They emulate human perception of video quality since they produce results which are very similar to those obtained from subjective methods. Most of these proposals were tested and compared in the different phases carried out by the video quality experts group (VQEG), which was formed to develop, validate, and standardize new objective measurement and comparison methods for video quality. The models that the VQEG forum validates result in International Telecommunication Union (ITU) recommendations and standards for objective quality measurement for both television and multimedia applications [1]. Some of the QAM proposals are designed to be as generalist as possible, that is, to be able to assess quality for a wide set of different

distortion types, while other QAM focus their design on the detection of one, two, or a reduced set of specific distortions.

It would be desirable to find a QAM for image and video that exhibits a good behavior for any set of video and/or image distortions, that is, that detects accurately (as close as possible to the human perceived quality) any distortion regardless of its type and degree. Also, it would be desirable that the time required to obtain a quality measurement is short enough in order to have a practical use or even to be able to use it in real time.

But quality is by definition a highly subjective feature that is influenced not only by the intrinsic characteristics of the signal but also by psychological and environmental factors. Therefore, the task of choosing "the best QAM" is influenced by too many factors and sources of inaccuracy. These sources of inaccuracy are, for example, the reliability of unbiased subjective reference data, the selection of video or image contents, the degree of the impairments and where they appear (in space and time), the procedure used to map between subjective and objective quality values, and even the use and interpretation of the correlation indicators. These factors must be taken into account when making comparisons between metrics [2].

The selection of a QAM may also depend on the target application where it will be used. Examples of applications are a real-time monitor that adaptively adjusts the image quality in a video acquisition or transmission system, a benchmarking image processing system, or even algorithms and encoder proposals that are embedded into image processing systems to decide about the preprocessing and postprocessing stages.

We work with a set of the most relevant quality assessment metrics whose source code or test software has been made available by their authors. So, we can use them in our own evaluation tests.

As mentioned before, we will analyze the behavior of the candidate metrics in two test environments. The first one, is the compression environment, where the quality of compressed sequences at different bitrates with different encoders is compared by means of QAM. The most common way of doing the comparisons between existing image/video coding approaches, improvements over these approaches, or completely new codec designs is by performing a rate/distortion (R/D) analysis. When using R/D, the distortion is usually measured in terms of PSNR (peak signal-to-noise ratio) values, where rates are often measured in terms of bpp (bits per pixel) for images or bps (bits per second) for video. So, in this test environment, we work with the selected QAM as candidates to replace the PSNR as the distortion metric in the R/D comparisons. We will also consider the QAM complexity in order to determine their applicability. The second one is the packet loss environment, where we will analyze the behavior of the candidate metrics in the presence of packet losses under different mobile ad hoc networks (MANET) scenarios. In particular, we are going to compare the behavior of QAM when measuring the quality degradation of an H.264/AVC video delivery in a MANET. We use a hidden Markov model (HMM) to accurately reproduce the packet loss patterns typical of these networks, including variable network congestion levels and different degrees of node mobility. For each particular network scenario, we perform a bitstream erasure process based on the loss patterns suggested by the HMM model. The resulting bitstream is delivered to the H.264/AVC decoder in order to get the resulting HRC that will be used to calculate the QAM value.

The organization of the paper is as follows. In the next section, we will describe the main frameworks addressing objective QAM. In Section 3, we will expose some key aspects of how to compare heterogeneous metrics and the method used to compare the metrics under evaluation. In Section 4, we show the behavior of several available quality metrics in the compression environment. In Section 5, the models and the methods used for the packet loss environment are explained and a behavioral analysis of the metrics is made for different network scenarios. Finally, in Section 6, we present the main conclusions of this work.

2. Objective Quality Assessment Metrics

In the past years, a big effort has been done in the field of QAM. A large number or objective metrics can be found in the literature. Some of them have been designed for a specific kind of distortions, while others are more generalist and try to assess quality regardless of the distortion type. Besides, each metric design is different. Objective evaluation of picture quality in line with human perception is still difficult [3–9] due to the complex, multidisciplinary nature of the problem, including aspects related to physiology, psychology, vision research, and computer science. Nevertheless, with proper modeling of major underlying physiological and psychological phenomena and by obtaining results from psychophysical tests and experiments, it is possible to develop better visual quality metrics to replace nonperceptual criteria as PSNR or MSE being still widely used nowadays.

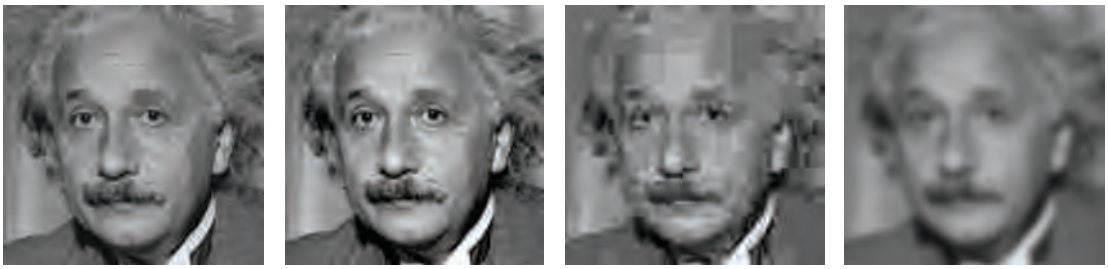
In the literature, we can find different classifications and frameworks that group several QAM depending on the way they are designed. In this section, we will briefly describe the main ideas behind the different frameworks, along with their main QAM.

There is a consensus in a primer classification of objective quality metrics [10, 11] attending to the availability of original nondistorted info (video reference) to measure the quality degradation of available distorted versions.

(i) Full reference (FR) metrics perform the distortion measure with full access to the original image/video version, which is taken as a perfect reference.

(ii) No reference (NR) metrics have no access to the reference image/video. So, they have to perform the distortion estimation based on the distorted version only. In general they have lower complexity but are less accurate than FR metrics and are designed for a limited set of distortions and video formats.

(iii) Reduced reference (RR) metrics have access to partial information about the original video. A RR metric defines what information has to be extracted from original video, so it can be compared with the the same one extracted from the distorted version.



(a) (b) (c) (d)

Figure 1: Example of three figures with different impairments and the same PSNR values: (a) original, (b) contrast stretched 26.55 dB, (c) JPEG compressed 26.60 dB, and (d) blurred 26.55 dB.

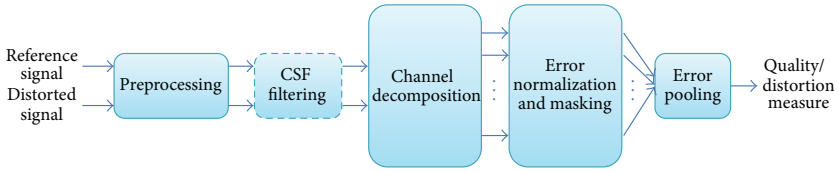


Figure 2: Common block diagram of the error sensitivity framework.

The most widely used FR objective video quality metrics are the mean square error (MSE) and the peak signal-to-noise ratio (PSNR). They are simple and quick to calculate, providing a good way to evaluate the video quality [12]. However, it is well known that these metrics do not always capture the distortion perceived by the human visual system (HVS). In Figure 1, an original image has been distorted in different ways. The PSNR metric gives almost the same value for each distortion, indicating that the quality of the distorted images is the same, but as it can be seen, the perceived quality is different for each image. Moreover, it is not unusual that the perceived quality of image in Figure 1(b) is higher than the one given to the original one, Figure 1(a). That is, a distorted image has better perceptual quality than the original one. If PSNR is used for measuring the quality of the resulting images/videos produced by the different coding proposals, how can we certify that one coding proposal has a better perceptual quality than another?

In this section, we will briefly describe also the main ideas behind the different frameworks and the most relevant and cited QAM of each one. QAM can be classified by many factors such as the metric architecture (number and type of blocks and stages or algorithms used in the metric design), the primary domain (space or frequency) where they work, and the inclusion or not of HVS characteristics or HVS models in their design.

2.1. HVS Model Based Framework. A basic idea of any metric based on a HVS model is that subjective differences between two images cannot be extracted directly from the given images (original and distorted one) but from their perceived versions, that is, from the version that our brain perceives. As it is known, the HVS produces several visual scene information reductions, carried out in different steps. The way in which this information reduction process is modeled is the key to obtain a good subjective fidelity metric.

This framework includes the metrics that are clearly based on a HVS model, that is, their design follow the stages of any of the available HVS models. We include here metrics from the error sensitivity framework (ESF) [7] and also some other RR and NR metrics that are based on HVS models.

This framework mainly include FR metrics based on HVS models that measure errors between the reference and the distorted content using a HVS model.

In general, the emulation of HVS is a bottom-up approach that follows the first retina processing stages to continue with different models of the visual cortex behavior. Also, some metrics deal with cognitive issues about the human visual processing modeling that are included as additional stages.

The main difference between the FR metrics of this framework is related to the way they perform the subband decomposition inspired by the complex HVS models [13–15], low cost decompositions in DCT [16, 17] or wavelet [18] domains, and with other HVS related issues like in [19] where foveal vision is also taken into account and in [20] where focus of attention is also considered. It is worth noting that most of proposed FR quality assessment models share the error sensitivity based philosophy which is motivated from psychophysical vision science research [11].

Figure 2 shows a block diagram with the typical processing stages of a FR metric. In the preprocessing stage, different operations are done in order to adequate some characteristic of the reference and the distorted input versions. These operations commonly include pixel alignment, image cropping, color space transformations, device calibrations, PSF filtering, light adaptation, and other operations. Not all the metrics perform all these operations; each metric processes both signals in a different way. After the preprocessing stage,

usually HVS models first decompose the input signal into spatiotemporal subbands at both the reference and distorted signals.

The contrast sensitivity function (CSF) can be implemented in the channel decomposition step by the use of linear filters that approximate the frequency responses to the CSF like in [21]. But most of the metrics choose to implement the CSF as weighting factors that are applied to the channels after the channel decomposition, providing for each channel a different perceptual sensitivity.

As mentioned before, frequency decomposition is one of the biggest differences between models and hence between metrics. Complex HVS frequency channel decomposition models are used in QAM designs, but some of these models are simplified attending to computational constraints. In this sense, other QAM use the DCT [16] or wavelet [18] transforms showing good MOS correlation results. Depending also on the metric type and the distortions it handles, metrics use different channel decomposition models.

Cortical receptive fields are represented by 2D Gabor functions, but the Gabor decomposition is hard to compute and is not suitable for some operations as invertibility, reconstruction by addition, and so forth. In [22], Watson modeled a frequency and orientation decomposition with profiles similar to the 2D Gabor functions but computationally more efficient. Other authors like Lubin [23], Daly [24], Teo and Heeger [13], and Simoncelli et al. [25] provided different models trying to approximate as close as possible the HVS channel decomposition.

There are also some models that use temporal frequency decomposition in order to account for the characteristics of the temporal mechanisms in the HVS [21, 26]. The design of temporal filter banks is typically implemented using infinite impulse response filters (IIR) with a delay of only a few frames; other authors use finite response filters that despite their higher delay are simpler to implement.

The next step is error normalization and masking. Masking occurs when a stimulus that is visible by itself cannot be detected due to the presence of another stimulus. In contrast, facilitation occurs when a nonvisible stimulus becomes visible due to the presence of another stimulus. Most of the HVS models implement error normalization and masking as a gain-control mechanism, using the contrast visibility thresholds to weight the error signal at each channel. Some metrics [14], due to complexity and performance issues, use only intrachannel masking, while others [13] include interchannel masking, as there are evidences that channels are not totally independent in the HVS. Other authors [27] include also in this stage the luminance masking, also called light adaptation. In [28, 29], some comparisons of different masking models and some considerations about how to include them into an image encoder are made. In [30], authors propose a contrast gain-control model of the HVS that incorporates also a contrast sensitivity function for multiple oriented bandpass channels.

The last processing step (Figure 2) is the error pooling, which is in charge of combining the error signals in different channels into a single distortion/quality interpretation, providing different importance to errors depending on the channels where they appear. For most QAM, a Lp norm or Minkowski norm is used to produce an image spatial error maps. From the spatial error map, a frame-level distortion score is computed. In video quality assessment, we obtain the corresponding sequence-level distortion score by averaging frame scores. For the time domain, some metrics use temporal HVS models or information to accurately reproduce human scores, while others simply do not assess time domain. Other QAM that may be included in the model based framework may be found in [13, 15–21, 26, 27, 31–36].

2.2. HVS Properties Framework. In this framework we consider the metrics that, although are not based on a specific HVS model, are still inspired in features of the HVS. We also include those metrics that are designed to detect specific impairments produced by any of the processing stages of image and video coding, like quantization, transmission errors, and so forth.

The Institute for Telecommunication Sciences (ITS) presented in [37] an objective video quality assessment system that was based on human perception. They extract several features from the original and degraded video sequences that were statistically analyzed in comparison with the corresponding human rating extracted form subjective tests. This analysis provide parameters to adjust objective measures for these features and, after being combined in a simple linear model, they provide the final predicted scores. Some of the extracted features require the presence of the original sequence, while others are extracted in a no-reference mode. The proposed metric exploits spatial and temporal information. The processing include Sobel filtering, Laplace filtering, fast Fourier transforms, first-order differencing, color distortion measures, and moment calculation.

In [38], authors proposed a RR metric for in-service quality monitoring system. Their metric is built on a set of spatiotemporal distortion metrics that can be used for monitoring in-service of any digital video system. Authors expose that a digital video quality metric, in order to be widely applicable, must accurately emulate subjective responses, must work over the full range of quality (from very low bit rate to very high), must be computationally efficient, and should work for end-to-end in-service quality monitoring. The metrics are based on extracted features from the video sequence as in [37] and in order to satisfy the last condition (to be able to work in in-service monitoring systems), these features, extracted from spatiotemporal regions are sent, compressed following the ITU-R Recommendation BT.601, through an ancillary data channel so that it can be continuously transmitted. In the paper, the authors describe these spatiotemporal distortion metrics in detail, so that they can be implemented by researchers.

Later, through The National Telecommunications and Information Administration (NTIA), the same authors, proposed the general model of the video quality measurements techniques (known as VQM metric [39, 40]) for estimating video quality and its associated calibration techniques. This metric was submitted to be independently evaluated on MPEG-2 and H.263 video systems by the video quality experts group (VQEG) in their phase II full reference

television (FR-TV) test. In [41], authors reduce the requirements of some of the features extracted in the NTIA general model in order to achieve a monitoring system that uses less than 10 kbits/s of reference information.

We also can find metrics based on watermarking techniques that analyze the quality degradation of the embedded image [42]. There are metrics that are designed for measurement-specific distortions types and those produced by specific encoders [43, 44]. Another representative metrics in this framework are the ones proposed in [43–49].

2.3. Statistics of Natural Images Framework. Some drawbacks of the model based HVS framework are reviewed in [7, 50]. Some of these drawbacks are, for example, that the HVS models work appropriately for simple spatial patterns, like pure sine waves; however, when working with natural images, where several patterns coincide in the same image area, their performance degrades significantly. Another drawback is related to the Minkowski error pooling, as it is not a good choice for image quality measurement. As authors show, different error patterns can lead to the same final Minkowski error.

Therefore, several authors argue that the approach to the problem of perceptual quality measurement must be a top-down approach, analyzing the HVS to emulate it at a higher abstraction level. The authors supporting this approach propose using the statistics of the natural images. Some of them propose the use of image statistics to define the structural information of an image. When this structural information is degraded, then the perceptual quality is also degraded. In that sense, a measurement of the structural distortion should be a good approximation to the perceived image distortion. These metrics are able to distinguish between distortions that change the image structure and distortions that do not change it, like changes in luminance and contrast.

In [7, 51], authors define a Universal Quality Index that is able to determine the structural information of the scene. This index models any distortion as a combination of three different factors: (a) the loss of correlation between the original signal and the distorted one, (b) the mean distortion that measures how close the mean of the original and distorted version are, and (c) the variance distortion that measures how similar the variances of the signals are. The dynamic range of the Quality Index is $[-1, 1]$. A value of 1 indicates that both signals are identical. They apply this index in a 8×8 window for an image, obtaining a quality map of the image. The overall index is the average of the quality map.

Authors in [50] further improve their previous quality index and in [52] propose a generalization of their work where any distortion may be decomposed into a linear combination of different distortion components. In [53], the model is extended to the complex wavelet domain in order to design a robust metric to scaling, rotation, and translation effects.

Authors in [54] proposed a video quality metric following a frame by frame basis. It takes quality measures for different blocks of each frame taking into account their spatial variability, the movement, and other effects (like blocking) by means of a specifically adapted NR metric [45].

Other authors use also statistics of the natural scene in a different way. They state that the statistical patterns of natural scenes have modulated the biological system, adapting the different HVS processing layers to these statistics. First a general model of the natural images statistics is proposed. The modeled statistics are those captured with high quality devices working in the visual spectrum (natural scenes). So, text images, computer generated graphics, animations, draws, random noise or image, and videos captured with nonvisual stimuli devices like radar, sonar, X-ray, and so forth are out of the scope of this approach. Then, for a specific image, the perceptual quality is measured taking into account how far its own statistics are from the modeled ones. In [55], a statistical model of a wavelet coefficient decomposition is proposed, and in [56] the authors propose an NR metric derived from previous work.

Some metrics defined under this approach take the objective quality assessment as an information loss problem, using techniques related to information theory [57, 58].

2.4. Metrics under Study. Now, we introduce the metrics we will use in our study. The criteria to choose these metrics, and no other ones, was the availability of their code (source or executable) to reproduce their behavior as follows.

(i) The DMOSp-PSNR metric: we translate the traditional PSNR to the DMOS space applying a scale-conversion process. We call the resulting metric DMOSp-PSNR.

(ii) The Mean Structural SIMilarity index [50] (MSSIM) from the structural distortion/similarity framework: in the reference paper, this FR metric was tested against JPEG and JPEG2000 distortion types. We test its performance with the new distortion types available in the second release of Live Database, "Live2 Database" since it is considered a generalist metric.

(iii) The visual information fidelity (VIF) metric [59] from the Statistics of Natural Images Framework. A FR metric that quantifies the information available in the reference image and determine how much of this reference information can be extracted from the distorted image.

(iv) The no-reference JPEG2000 quality assessment (NRJPEG2000) [54] from the Statistics of Natural Images Framework. A NR metric that uses natural scene statistical models in the wavelet domain and uses the Kullback-Leibler distance between the marginal probability distributions of wavelet coefficients of the reference and distorted images as a measure of image distortion.

(v) Reduced-reference image quality assessment (RRIQA) [57] from the Statistics of Natural Images Framework. The only RR metric under study. It is based on a natural image statistical model in the wavelet transform domain.

(vi) The no-reference JPEG quality score (NRJPEGQS) [43] from the HVS properties framework. A NR

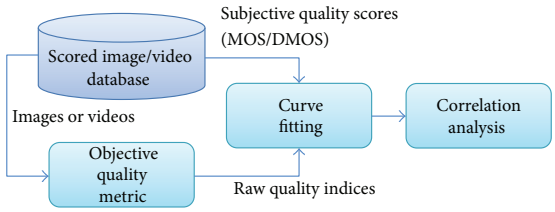


Figure 3: Block diagram of the QAM evaluation process.

metric designed specifically for JPEG compressed images.

(vii) The video quality metric [40] (VQM general model) from the HVS properties framework. The VQM uses RR parameters sent through an ancillary channel that requires at least 14% of the uncompressed sequence bandwidth. Although being conceptually an RR metric, it was submitted to the VQEG FR-TV test because the ancillary channel can be used to receive more detailed and complete references from the original frames, even the original frames themselves.

3. Comparing Heterogeneous Metrics

As previously mentioned, each QAM gets the quality of the image/video using its own and specific scale that depends on its design. Therefore, these raw quality scores cannot be compared directly, even though the range of the values (scale) is the same. In order to compare fairly the behavior of various metrics for a set of images or sequences, the objective quality index obtained from each metric has to be converted into a common scale.

When reviewing the performance comparisons that authors made in their new QAM proposals, few details are provided about the comparison procedure itself. So, it is difficult to replicate these results. Authors in [2] reviewed the sources of inaccuracy of each step of the QAM comparing process, shown at Figure 3. The sources of inaccuracy may be related to many factors as the reliability of the subjective reference data, the types and grade of the distortions in the images or videos, the selection of the content that made up the training and testing sets, and even the use and interpretation of the correlation indicators. These sources of inaccuracy can lead to quantitative differences when the same QAM is tested by different authors, even when the tests are correctly done. Although different tests can provide slightly varying results for a set of metrics, their results should be in line as explained in [2].

These issues encouraged and guided us to perform our own comparison test with the selected QAM in order to adapt the test to the target applications we are interested in. The results of our test, as expected, were slightly different from other comparison tests but remain in line with their results [2].

We use the method and mapping function proposed by the VQEG [6, 60] with some refinements proposed in other relevant comparison tests [61]. The chosen target scale is the DMOS scale (differences mean opinion score) which is the one used by the VQEG and other authors [61] when comparing metric proposals.

In order to compare several QAM, first a subjective test must be done, for example, a Double Stimulus Continuous Quality Scale (DSCQS) method as suggested and explained in [6], in order to get the subjective quality assessment of a set of images or sequences. The scale used by the viewers goes from 0 to 100. Raw scores obtained in subjective tests are converted into difference scores and processed further [58] to get a linear scale in the 0–100 range. The mean opinion score (MOS) can be calculated for the source and distorted versions of each image or sequence in this set. The DMOS is therefore the difference between the MOS value obtained for the original image/sequence and the MOS value obtained for the distorted one. So, for a particular image or sequence, its DMOS value gives the mean subjective value of the difference between the original and the distorted versions. A value of 0 means no subjective difference found between the images by all the viewers. Due to the nature of the subjective test this value is very unlikely.

In this work, we have not done such a subjective test. Instead of this, we have used directly the DMOS values published in the Live Database Release 2 [62] and in the VQEG Phase I Database [63].

Basically, the raw score of each metric must be converted into a value in this predicted DMOS (DMOSp) scale. This is done in the curve fitting step, shown in Figure 3. The final result of this scale-conversion process allows the quality score given by a metric for a specific image/sequence to be directly comparable with the one given by the other metrics for the same image/sequence.

We use the nonlinear mapping function between the objective and the subjective scores, as suggested in the VQEG Phase I and Phase II testing and validation tests [6, 60] as well as in other extensive metrics comparison tests [61]. This function is shown in (1). It is a parametric function which is able to translate a QAM raw score to the DMOSp space. As suggested in [2, 64], the performance evaluation of the metrics (correlation analysis step in Figure 3) is computed after a nonlinear curve fitting process.

A linear mapping function cannot be used because quality scores are rarely scaled uniformly in the DMOS scale, because different subjects may interpret vocabulary and intervals of the rating scale differently, depending on the language, viewing instructions, and individual psychological characteristics. Therefore, a linear mapping function would give too pessimistic view of the metric performance. Several mapping functions could be selected for this purpose, such as cubic, logistic, exponential, and power functions, with monotonicity being the main property that the function must comply with, at least in the relevant range of values.

Consider

$$\text{Quality}(x) = \beta_1 \text{logistic}\left(\beta_2, (x - \beta_3)\right) + \beta_4 x + \beta_5, \quad (1)$$

$$\text{logistic}(\tau, x) = \frac{1}{2} - \frac{1}{1 + \exp(\tau x)}. \quad (2)$$

Table 1: Equation parameters of metrics under study.

	β_1	β_2	β_3	β_4	β_5
MSSIM	−39.5158	14.9435	0.8684	−10.8913	46.4555
VIF	−3607.3040	−0.5197	−1.6034	−476.0144	−693.3585
NRJPEGQS	37.6531	−0.9171	6.6930	−0.2354	40.7253
NRJPEG2000	37.3923	0.8190	0.6011	−0.8882	74.5031
RRIQA	−18.9995	1.5041	3.0368	6.4301	5.0446
PSNR-DMOSp	23.2897	−0.4282	28.7096	−0.6657	61.5160
VQM-GM	−163.6308	6.3746	−7.6192	114.4685	76.6525

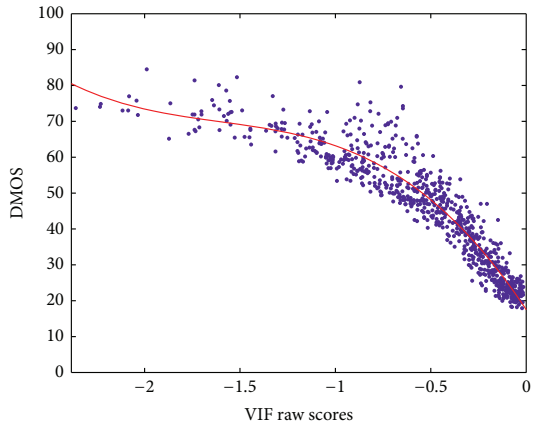


Figure 4: Dispersion plot used for the VIF metric including the curve fit for (1).

Equation (1) has five parameters, from β_1 to β_5, that are fixed by the curve fitting process that achieves the best correlation between the QA metric values and the subjective DMOS values. We have not found in the literature any mapping function with its parameters for any image/video database. So, we have calculated these parameters based on sets of images and sequences that conform with our "training sets".

As an example, Figure 4 shows the dispersion plot used in the fitting process for one of the metrics, in this case the VIF metric. Each point of the scatter-plot corresponds to an image of the training set used, Live2 Database [62]. For each image in the training set, we get the average DMOS value obtained in the subjective test and we run each metric in order to get its raw quality scores. Each metric gives its score in its own scale.

The x-axis of Figure 4 corresponds to the raw values given by the VIF implementation used, where 0 corresponds to the highest quality reported by the metric and decreasing values report lower quality. In the y-axis, we have the corresponding DMOS values. The curve fitting process gives us the parameters for (1), which is represented by the solid curve in Figure 4.

The quality of the images in the subjective test is variable, covering a large range of distortion types and intensities for each distortion. Image distortions go from very highly distorted to practically undistorted ones. The viewers gave their scores for each image in the set, obtaining the average DMOS value. As shown in Figure 4, the dynamic range of the average DMOS values does not reach the limits of the DMOS scale (0 and 100) for any distortion type; therefore, the fitted curve predicts DMOSp values inside the same dynamic range. This is the reason why for a raw score of 0 (the best possible quality for the metric in this case), the predicted DMOSp value is not 0; that is, there was no image scored with an average DMOS value of 0, instead of that, the best DMOSp value obtained is around the value of 20. So, in the case of the VIF metric its dynamic DMOSp range varies from 20 to 80.

Table 2: Goodness of DMOSp-DMOS fitting.

	PCC	RMSE	SROCC
MSSIM	0.8625	7.9682	0.851
VIF	0.9529	0.0516	0.9528
NRJPEGQS	0.936	3.0837	0.902
NRJPEG2000	0.9099	7.056	0.9021
RRIQA	0.9175	4.4986	0.9194
PSNR-DMOSp	0.85257	9.0969	0.8197
VQM-GM	0.8957	7.6746	0.9021

Having fixed the beta parameters for each metric (see Table 1), (1) can be used to estimate or predict the DMOSp value for any objective metric score.

In Table 2, the performance of our fittings is shown. These performance parameters show the degree of correlation between the DMOSp values and the subjective DMOS values provided by the viewers. Performance validation parameters are the Pearson correlation coefficient (PCC), the root mean squared error (RMSE), and the Spearman rank order correlation coefficient (SROCC).

Another key point to consider while comparing QAM [2] is the selection of the image or video sequence set used as "training set." The "training set" is used to perform the curve fitting process. This set should be chosen with special care and must be excluded from validation tests. So for each metric, the fitting process must be done using images or sequences with impairments that the metric is designed to handle. See [2] for details of how an incorrect selection of the image "training set" can influence the final interpretation of the statistics used in the correlation analysis.

Once the metric has been evaluated in the correlation analysis step, it will work with another set of images or sequences that we call the "*testing set*." For the "testing set," the DMOS values are unknown; therefore, we obtain them via (1).

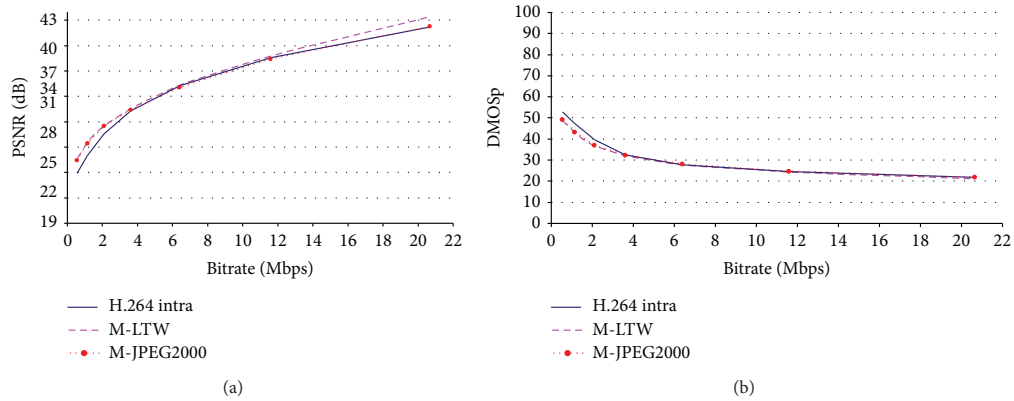


Figure 5: PSNR versus DMOSp-PSNR for the evaluated codecs (mobile sequence).

In our study, all the metrics have been "trained" only with the luminance information. The MSSIM, VIF, RRIQA, and DMOSp-PSNR metrics were "trained" with the whole Live2 Database because they are intended to be generalist metrics.

The NRJPEGQS was "trained" only with the JPEG distorted images of Live2 database as this metric is designed only to handle this type of distortions. And for the same reason the NRJPEG2000 was "trained" only with the JP2 K distorted images of Live2 Database and the VQM-GM was "trained" with a subset of 8 video sequences and its 9 corresponding HRCs of the VQEG Phase I Database in a bitrate range of 1 to 4 Mb/s.

It is important to mention that each of these "training sets" has different dynamic ranges in the DMOS scale depending on the degree of distortions applied to the images in each set.

We define as "homogeneous metrics" those which were trained with the same sets, and therefore, we use the term "heterogeneous metrics" to refer to metrics that were trained with different sets.

Our "testing set" comprises different standard video sequences that are commonly used in video coding evaluation research, as shown in Table 3. For FR-metrics, both reference and distorted images/sequences are used as input. For NR-metrics only the distorted image/sequence is available. For RR-metrics, the reference image/sequence is the input of the features extraction step, and both the extracted features and the distorted image/sequence are the input for the final metric evaluation step. Image metrics were applied to each frame of the sequences and the mean raw value for all the frames was translated to the DMOSp scale. Hence, we finally obtain comparable DMOSp values for all images/sequences.

4. Analyzing Metrics Behavior in a Compression Environment

In this section, we will study the behavior of the QAM under evaluation when assessing the quality of compressed images and sequences with different encoders. As exposed before, in the development of a new encoder or when performing modifications to existing ones, the performance of the proposals must be evaluated in terms of perceived quality by means of the R/D behavior of each encoder. The distortion metric commonly used in the R/D comparisons is PSNR.

Table 3: Sequences included in the "test set".

Sequence	Frame	F. number	F. rate
Foreman	QCIF: 176 × 144	300	30 fps.
Container			
Foreman	CIF: 352 × 288		
Container			
Mobile	640 × 512	40	

So, in this test environment, we will work with the selected metrics as candidates to replace the PSNR as the quality metric in a R/D comparison of different video codecs. In this case, we will use a set of video encoders and video sequences in order to create distorted sequences hypothetical reference circuit (HRC) at different bitrates and analyze the results of the different QAM under study. Also, we will consider the metric complexity in order to determine their scope of application. For the tests, we have used an Intel Pentium 4 CPU Dual Core 3.00 GHz with 1 Gbyte RAM. The programming environment used is Matlab 6.5 Rel.13. The codecs under test are H.264/AVC [65], Motion-JPEG2000 [66], and Motion-LTW [67]. The fitting between objective metric values and subjective DMOS scores was done using the Matlab curve fitting toolbox looking for the best fit in each case.

A R/D plot of the different video codecs under test, using the traditional PSNR as a distortion measure, is shown in Figure 5(a). It is usual to evaluate performance of video codecs in a PSNR range varying from 25–27 dB to 38–40 dB, because it is difficult to determine which one is better with PSNR values above 40 dB. This saturation effect, at high qualities, is not captured by the traditional PSNR that increases steadily as the bitrate rises, as shown in Figure 5(a).

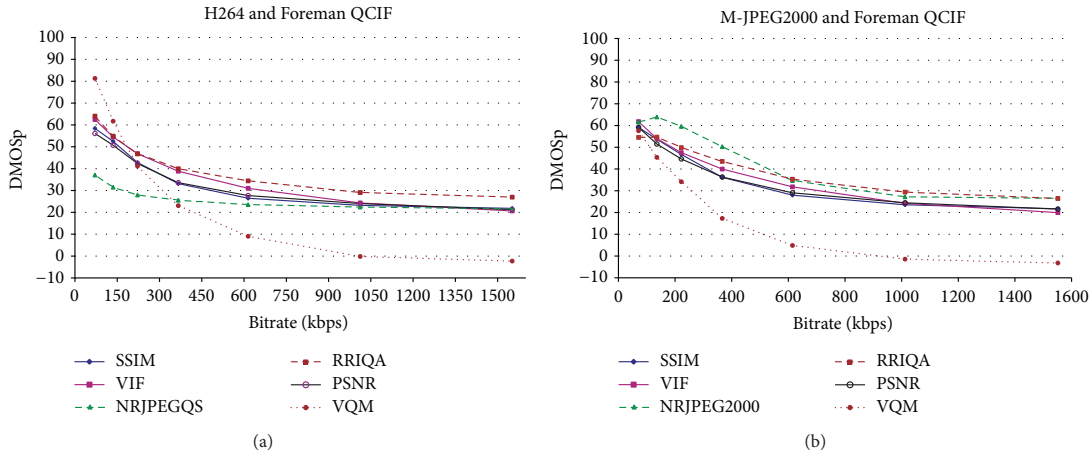


Figure 6: QAM comparison using the same sequences with different codecs.

We convert the traditional PSNR to a metric that we call DMOSp-PSNR by applying the scale-conversion process explained in Section 3. We can consider the DMOSp-PSNR metric to be the "*subjective*" counterpart of the traditional PSNR. It is the same metric, though expressed in a different scale. The DMOSp scale denotes distortion, thereby quality increases as DMOSp value decreases. The main difference between PSNR and its counterpart DMOSp-PSNR is that the saturation effect is fixed, as we can see in Figure 5(b). As it can be seen, subjective saturation effect is noticeable above a specific quality value. At bitrates above 11.5 Mbps, the DMOSp values practically do not change. This behavior is the same for all the evaluated codecs and video formats, confirming that there is no noticeable subjective difference when watching the sequences at the two highest evaluated bitrates (11.7 and 20.7 Mbps).

But as mentioned before the only modification that has been done to the PSNR metric was the mapping process with the DMOS data; that is, the raw values of the PSNR have not changed; therefore, DMOSp-PSNR metric does not fix the known drawbacks shown in Figure 1. For bitrates values below the saturation point (11.5 Mbps in the case of Figure 5(b)), the behavior of the two R/D curves is the same. In fact, the DMOSp-PSNR metric below the saturation point arranges the codecs by quality in the same order as the PSNR does, agreeing also with the results of subjective tests. This behavior is the same for all evaluated sequences and bitrates.

Since PSNR, and therefore DMOSp-PSNR, are known to be inaccurate perceptual metrics for image or video quality assessment, we now analyze the remaining metrics under study for all codecs and bitrates. These metrics have a better perceptual behavior and they offer different scores for the images in Figure 1.

The expected behavior of a QAM scoring an image or sequence at different bitrates is as follows.

(i) It should give a decreasing quality value as the bitrate decreases when bitrate values are below saturation threshold.

(ii) The quality value should be almost the same when bitrate values are above saturation threshold.

So, we run all the metrics for each HRC and analyzed the resulting data between consecutive bitrates, obtaining the quality scores in the DMOSp space. A simple subjective DSCQS test was performed with 23 viewers in order to detect if there was really perceived differences above threshold in these sequences at high bitrates (above saturation 11.5 Mbps). In the tests, the three HRCs (for each sequence and encoder) with higher bitrates were presented to the viewers: the first HRC (the first located below saturation point, 6.4 Mbps) and the last two HRCs (two rightmost points from curves in Figure 5, 11.58 and 20.65 Mbps) that are locate in the saturation region. The test concluded that no perceptual differences were detected above saturation threshold, whereas all the viewers detected some perceptual differences bellow threshold. The predicted DMOSp differences for these HRCs above threshold vary from 0.82 to 4.91 DMOSp points, so we can initially conclude that above saturation these small differences in DMOSp values are perceptually indistinguishable.

In Figure 6 we can see examples of the R/D plots used for comparing the metrics where all the evaluated QAM were applied to the same sequence. In Figure 6(a), the HRCs were encoded with the H.264/AVC codec. The NRJPEG2000 metric is omitted because it is not designed to handle DCT transform distortions. In the same way, in Figure 6(b), where HRCs were encoded with M-JPEG2000, the NRJPEGQS metric is omitted because it is not designed to handle the distortions related to the wavelet transform. We can see that the perceptual saturation is captured by all the QAM at high bitrates (high quality) regardless of the encoder. The same holds for all the sequences and encoders.

As mentioned in Section 3, monotonicity is expected in the mapping function. So, the expected behavior of the metrics should also be monotonic; that is, metrics should indicate lower quality values as the bitrates decrease. However, if we look at Figure 6(b) and focusing on the two lowest bitrates, the quality score given by both the RRIQA and

Figure 7: First frame of Foreman QCIF encoded at 70 Kbps (left) and 135 Kbps (right).

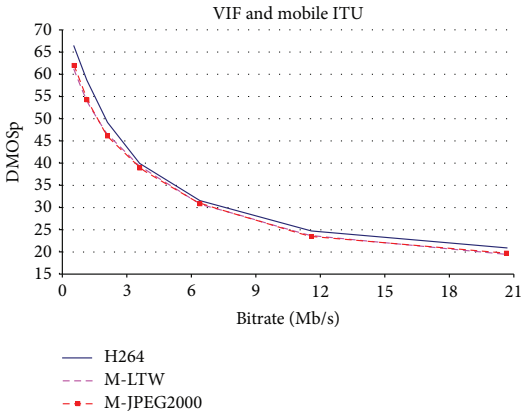


Figure 8: R/D performance evaluation of the three video codecs using mobile ITU video sequence by means of VIF metric.

NRJPEG2000 metrics increases as the bitrate value decreases. This is contrary to the expected behavior of a QAM. Figure 7 shows the first frame of the Foreman QCIF frame size sequence at these bitrates. Clearly, the right image (135 Kbps) receives a better subjective score than the left one (70 Kbps), though the mentioned metrics state just the opposite in this particular case. Our results for the compression environment show that NRJPEG2000 offers wrong quality scores between the two highest compression ratios with the M-JPEG2000 codec, for all the sequences and frame sizes tested. RRIQA also failed with this codec at high compression ratios, but only for small video formats. All the other metrics exhibit a monotonic behavior for all bitrates regardless of the encoder and sequence being tested.

Figure 6 will also help us to explain what it was exposed in Section 3; heterogeneous metrics should not be compared directly because the dynamic range of the subjective quality scores in each training set is different. Looking at Figure 6(a) and focusing on the lowest bitrate, the DMOSp rating differences between metrics arrive surprisingly up to 44.21 DMOSp units.

In fact, there are three different behaviors corresponding to the use of three different training sets: VQM-GM was trained with VQEG sequences, NRJPEGQS was trained only with the JPEG distorted images, and the rest of the metrics trained with the whole set of distorted images in the Live2 Database. This is the main reason of these anomalous behaviors in Figure 6.

So, when including in the same R/D plot curves from different metrics it should be checked that the metrics are homogeneous in order to avoid misleading conclusions.

Determining how good a metric works depends on how good the metric predicts the subjective scores given by human viewers. This goodness of fit is measured in parameters like those of Table 2. Our performance validation data tells that the VIF metric is the one which best fits the subjective DMOS values among the metrics in the same "training set."

Figure 8 represents the common R/D plots used when comparing the performance of the encoders being tested. In this case the plot shows how the VIF metric evaluates the performance of the encoders. If the mapping function of the metrics was obtained with the same "training set," then the ranking order of the encoders should agree with the subjective ranking order for each bitrate being evaluated.

We performed a simple subjective test with 23 viewers in order to evaluate if we can trust the codec ranking; that is, for a specific bitrate, the metric should arrange the encoders by quality, in the same order that a human observer does. For each rate and sequence, the reconstructed sequence of each encoder was presented simultaneously to the subjects. The ordering of the three sequences varies for each HRC, so that the subjects had no knowledge about the encoder order. The subjects ranked the sequences by perceptual quality if no differences were detected between pairs of sequences; they also annotated this fact. After analyzing the users scores and removing outliers, the test confirms that the ranking order of the metrics was the same as the subjective ranking.

In the cases where viewers scored no perceptual difference between sequences, the metrics gave always values lower than 2.9 DMOSp units of difference between encoders. In this test, for slightly higher differences, for example, 3.11 DMOSp units at 2.1 Mb/s between H264/AVC and M-JPEG2000 in Figure 8, most of the viewers could see some perceptual differences between the sequences, since they ranked H264/AVC to have better perceptual quality than M-JPEG2000 and M-LTW.

In order to determine how much difference expressed in the DMOSp scale is perceptually detectable, deeper studies and subjective tests must be done. From our studies, we detect that the perceptual meaning of the difference depends on the point in the DMOSp scale where we are working. For example, for high quality (as stated before in previous tests), DMOSp value differences up to 4.91 DMOSp points were imperceptible; however, at lower quality levels, smaller differences (3.11) can be perceived.

Finally, Table 4 shows, for different frame sizes, the mean frame evaluation time and the evaluation time for the whole sequence needed by each metric to assess its raw quality value. Times for the two steps of RRIQA, features extraction (f.e.), and quality evaluation (eval.) have been separately measured. For a CIF sequence (calibration and colour conversion time is not included) the VQM-GM is faster than the other metrics, except NRJPEGQS and DMOSp-PSNR. DMOSp-PSNR is by far the less computationally expensive metric at all frame sizes. On the other hand, RRIQA and VIF are the slowest

Table 4: QAM average scoring times (seconds) at frame and sequence level.

	QCIF		CIF		640 × 512	
	Frame	Seq.	Frame	Seq.	Frame	Seq.
MSSIM	0.028	8.4	0.147	44.1	0.764	30.5
VIF	0.347	104.1	1.522	456.5	6.198	247.9
NRJPEGQS	0.01	3	0.049	14.6	0.201	8.1
NRJPEG2000	0.163	48.9	0.486	145.9	1.595	63.8
RRIQA (f.e.)	4.779	1433.7	6.95	2084.9	10.111	404.5
RRIQA (eval.)	0.201	60.2	0.635	190.6	2.535	101.4
DMOSp-PSNR	0.001	0.3	0.006	1.7	0.02	0.8

metrics (they run a linear multiscale, multiorientation image decomposition), although in our tests the VIF is the most accurate metric among the general purpose metrics.

5. Analyzing Metrics Behaviour in a Packet Loss Environment

Our objective in this section is to analyze the behavior of the candidate metrics in the presence of packet losses under different MANET scenarios. In order to model the packet losses in these error prone scenarios, we use a three-state hidden Markov model (HMM) and the methodology presented in [68]. HMMs are well known for their effectiveness in modeling bursty behavior, relatively easy configuration, quick execution times, and general applicability. So, we consider that they fit our purpose of accelerating the evaluation process of QAM for video delivery applications on MANET scenarios, while offering similar results to the ones obtained by means of simulation or real-life testbeds. Basically, by the use of the HMM, we define a packet loss model for MANET that accurately reproduces the packet losses occurring during a video delivery session.

The modeled MANET scenario is composed of 50 nodes moving in an 870 × 870 square meters area. Node mobility is based on the random way-point model, and speed is fixed at a constant value between 1 and 4 m/s. The routing protocol used is DSR. Every node is equipped with an IEEE 802.11g/e enabled interface, transmitting at the maximum rate of 54 Mbit/s up to a range of 250 meters. Notice that a QoS differentiated service is provided by IEEE 802.11e [69]. Concerning traffic, we have six sources of background traffic transmitting FTP/TCP traffic in the best effort MAC access category. The foreground traffic is composed by real traces of an H.264 video encoded (using the Foreman CIF video test sequence) at a target rate of 1 Mbit/s. The video source is mapped to the video MAC acess category.

We apply the HMM described above to extract packet arrival/loss patterns for the simulation traces and later replicate these patterns for testing. We describe two environments: (a) congestion related environment and (b) mobility related environment.

The congestion environment is composed of 6 scenarios with increasing level of congestion, from 1 to 6 video sources. The mobility environment is composed of 3 scenarios with only one video source, but with increasing degrees of node mobility (from 1 to 4 m/s).

For each of these scenarios, we get different packet loss patterns provided by the HMM that represents each scenario.

After an analysis of the packet losses, different patterns are defined as follows.

(i) Isolated small bursts represent less than 7 consecutive lost packets. As each frame is split in 7 packets at source, isolated bursts will affect 1 or 2 frames, but none of them will be completely lost. This error pattern is mainly due to network congestion scenarios, where some packets are discarded due to transitory high occupancy in the wireless channel or buffers at relaying nodes.

(ii) Large packet loss bursts. Large Bursts cause the loss of one or more consecutive frames. Large packet error bursts are typically a consequence of high mobility scenarios, where the route to the destination node is lost and a new route discovery process should be started. This will keep the network link in down state during several seconds, losing a large number of consecutive packets.

We have used the H.264/AVC codec adjusting the error resilience parameters to the values proposed in [70], so that the decoder is able to reconstruct sequences even when large packet loss bursts occur. H.264/AVC is configured to produce one I frame every 29 P frames, with no B frames and to split each frame in 7 slices, so we put each slice into a separate packet and encapsulate its output in RTP packets. As suggested in [70], we also force 1/3 of the macroblocks of each frame to be randomly encoded in intramode.

We have used the Foreman CIF seq. (300 frames at 30 fps) to build an extended video sequence by repeating the original one up to the desired video length. After running the encoder for each extended video sequence, we get RTP packet streams. Then, we delete from the RTP packet stream, those packets that have been marked as lost packets by the HMM model. This process simulates packet losses in the MANET scenarios, so a distorted bitstream will be delivered to the decoder. The decoder behavior depends on the packet loss burst type as follows.

(i) When an isolated small bursts appear, the decoder is able to apply error concealment mechanisms to repair

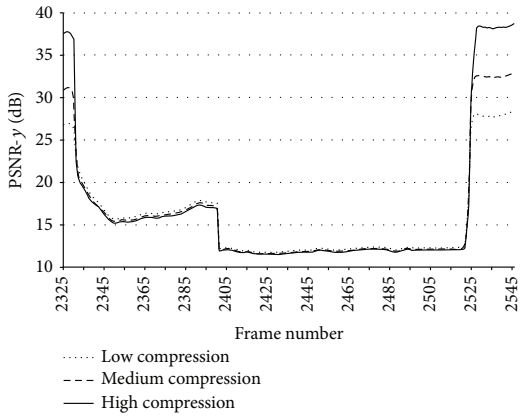


Figure 9: PSNR frame values during a long packet loss burst (from frame 2327 to 2525) at different bitrates.

the affected frames. The video quality decreases, and just after the burst, the reconstructed video quality recovers the quality by means of the random intra-coded macroblock updating. When the next I frame arrives, it completely stops error propagation.

(ii) When the decoder faces large bursts, it stops decoding and waits until new packets arrive. This produces a sequence in the decoder that is shorter than the original one. Therefore, both sequences are not directly comparable with the QAM and so we freeze the last completely decoded frame until the burst ends.

Once we have comparable video sequences (original and decoded video sequences with the same length), we are able to run the QAM. Each metric produces an objective quality value for each frame in its own scale. Then, we perform the scale-conversion to the DMOSp scale (see Section 3).

Figure 9 shows the objective quality value in the traditional PSNR scale at three different compression levels (low compression, medium compression, and high compression) during a large packet loss burst. We observe the evolution of quality during the burst period. What the observer sees during this large burst is a frozen frame, with more or less quality depending on the compression level. The PSNR metric reports that quality drops drastically with the first frame affected by the burst and decrease even more as the difference between the frozen frame and the current frame increases. Nearly at the middle of the burst, an additional drop of quality can be observed. It corresponds to a scene change (with the beginning of a new cycle of the foreman video sequence). At this point, the drastic scene change makes the differences between sequences even higher, and the PSNR metric scores with even worse values, reaching values as low as 10–12 dBs.

On the other hand, the perceived quality which changes at these levels is quite difficult to evaluate. So, a better perceptually designed QAM should not score such a quality drop in this situation because quality saturates. When the burst ends, quality rapidly increases because of the arrival of packets belonging to the same frame number than the current one in the original sequence (frame 2525 in Figure 9).

If during such a burst a QAM takes into account only the quality of the frozen frame, disregarding the differences with the original one (which changes over time), the effect of the burst would remain unnoticed for that metric, that is, quality remain constant.

Figure 10 shows the evolution of the candidate QAM during a large burst (similar to Figure 9 but in this case in the DMOSp space). There is a panel for each compression level: Figure 10(a) corresponds to high compression, Figure 10(b) to middle compression, and Figure 10(c) to low compression. We observe some interesting behaviors that we proceed to analyze.

From a perceptual point of view, quality must drop to a minimum when one or more frames are lost completely and should remain that way until the data flow is recovered. It should not matter if a scene change takes place inside the large burst. VIF and MSSIM behaves this way. At the point of the burst, where the scene change takes place, both the VIF and MSSIM metrics have almost reached their "bad quality" threshold regardless of the compression level and therefore there is no substantial change in the reported quality. The drop of quality to the minimum at the beginning of the burst evidence the lost of whole frames.

NR metrics do not detect the presence of a frozen frame (by dropping the quality score) as expected because the quality given by these metrics remain at the level scored for the frozen frame during the burst duration. So, NR metrics could not detect the beginning of a large burst, since lost frames will be replaced with the last correctly decoded frame (frozen frame) and the reference frames are not available for comparison. However, NR metrics detect the end of such bursts. Figure 11 will help us to explain this behavior, showing how reconstruction is done after a large burst. This figure shows the impairments produced when the large burst ends. Figure 11(a) is the current frame, the one being transmitted. Figure 11(b) is the frozen frame that was repeated during the burst duration. When the burst ends, the decoder progressively reconstruct the sequence using the intramacroblocks from the incoming video packets. So the decoder partially updates the frozen frame with the incoming intramacroblocks. This is shown in Figures 11(c) and 11(d) where the face of the foreman appears gradually.

The gradual reconstruction of the frame with the incoming macroblocks is interpreted in a different way by NR metrics and FR metrics. When the macroblocks begin to arrive, what happens at frame 2522 (see Figure 12), the NR metrics react scoring down quality, while the FR metrics begin to increase their quality score, just the opposite behavior. For a NR metric, without a reference frame, Figure 11(c) has clearly worse quality than Figure 11(b). But for a FR metric the corresponding macroblocks between Figures 11(c) and 11(a) help to increase the scored quality.

So, NR metrics react only when the burst of lost packets affects frames partially, that is, isolated bursts and at the end of a large burst. The NRJPEGQS metric reacts harder (i.e., it shows higher quality differences) than the NRJPEG2000

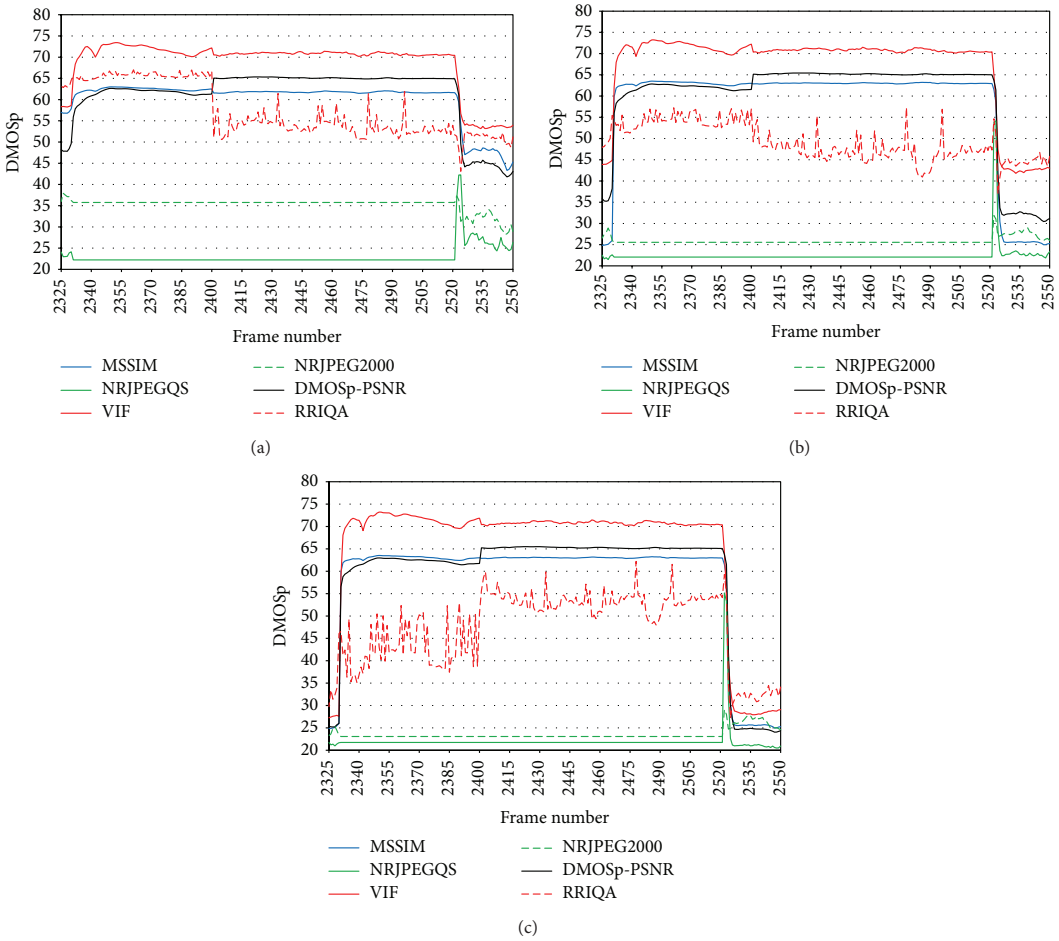


Figure 10: Metric comparison in the DMOSp space during a very large burst.

Figure 11: Frame reconstruction after a large burst: (a) original frame, (b) last frozen frame, and (c) (d) first and second reconstructed frames after the burst.

because it was designed to detect the blockiness introduced by the discrete cosine transform. When the frame is fully reconstructed then the score obtained with NR and FR metrics approaches again the values achieved before the burst, which depends on the compression rate.

The RRIQA metric shows high variability in its scores between consecutive frames inside bursts. These variations become more evident as the degree of compression decreases. The nature of the data sent through the ancillary channel, 18 scalar parameters obtained form the histogram of the wavelet

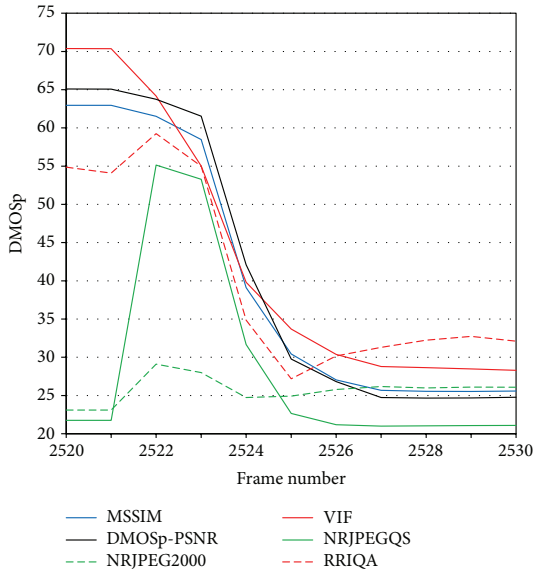


Figure 12: End of the large burst for the low compression panel. FR and NR metrics show the opposite behavior.

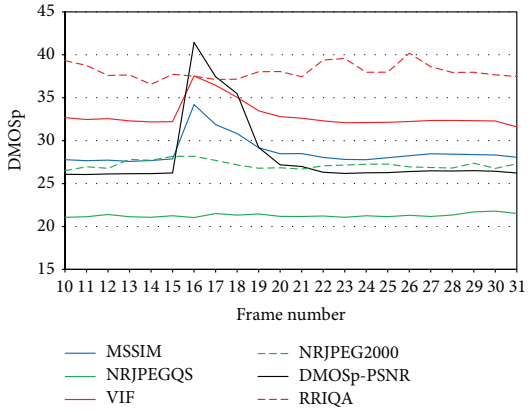


Figure 13: Metric comparison for an isolated burst.

subbands of the reference image, is very sensitive to loss of synchronism between the reference frame and the frozen one. On the decoder, the same extracted parameters are statistically compared with the received through the ancillary channel. When this comparison is performed with two sets of parameters obtained from different frames, unexpected results appear.

Concerning the FR metrics, MSSIM, VIF, and PSNR-DMOSp show a similar behavior or trend. MSSIM and PSNR-DMOSp show closer quality scores between them than the ones obtained with the VIF metric, which gives lower quality values than the other two metrics. This behavior is the same regardless of the compression level inside the large burst. Leaving aside the PSNR-DMOSp, which is not really a QAM, the other two FR metrics (VIF and MSSIM) have the same behavior when facing large bursts.

Figure 13 shows an isolated burst. In this case, blur and edge shifting impairments are introduced altering only one frame. This fact is perceived only by the FR metrics and the NRJPEG2000, which is designed to detect this type of impairments. The error concealment mechanism of H.264/AVC needs up to 6 frames to achieve the same quality scores obtained before the burst. Figure 14 shows the original frame (a) and three subsequent frames (b, c, d), where the effect of the lost packets is concealed by the H.264/AVC decoder.

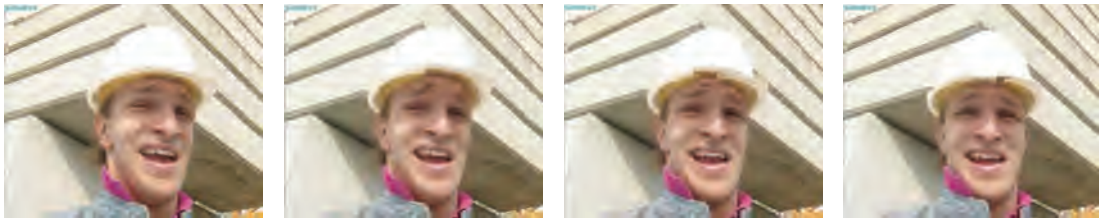
As defined previously, an isolated burst can affect one or two consecutive frames. In the last case, the behavior of the QAM when facing the isolated burst resembles the behavior of the metrics with a large burst. The difference is that the concealment mechanisms and the correct reception of part of the frames avoid the largest drop in the quality.

Figure 15 shows multiple consecutive bursts (large and isolated) that behave as exposed previously. From left to right, we see a large burst followed by an isolated one. This pattern repeats again one more time, and at the right most part of the figure, between frames 352 and 372, two large bursts occur consecutively, having a gap between them where new incoming packets arrive for a short period of time (frames 361 and 362).

In Figure 16, we zoom into this area (frames 352 to 372) to analyze why the behavior of the DMOSp-PSNR metric differs from the other FR metrics during the gap between bursts. In the gap, the encoder is not able to reconstruct a whole frame because the gap is too small, that is, between the two large bursts only a small amount of packets arrive, and this is not enough to reconstruct a whole frame. So the involved frames (361 and 362) are partially reconstructed (Figures 17(b) and 17(c)). Both frames exhibit perfect correspondence in the lower half with the original one (Figure 17(a)). Therefore, the scored quality must increase, at least to some extent, compared to the quality of the previous frozen frame, as occurs at the end of a large burst. This fact is only reflected by the VIF and MSSIM metrics. The PSNR-DMOSp metric is not able to detect this because it is computed using information from the whole frame. For the VIF and the MSSIM, which are perceptually driven, the lower half of the frame increases their raw scores, in the same way as the human scores do. After frame 362, quality decreases again since the following frame is frozen too. So, VIF and MSSIM detect two consecutive loss burst, while PSNR-DMOSp and the other metrics consider only a single larger one.

6. Conclusions

The main goal of this work was focused on looking for a quality assessment metric that could be used instead of the PSNR when evaluating compressed video sequences with different encoder proposals at different bitrates and to analyze the behavior of such metrics when compressed video is transmitted over error prone networks such as MANETs.



(a) (b) (c) (d)

Figure 14: Packet loss affecting only one frame. (a) Original frame and (b, c, d) next three decoded frames.

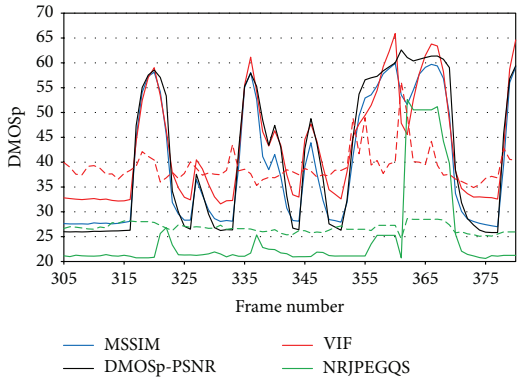


Figure 15: Frame interval where different type of bursts occurs consecutively.

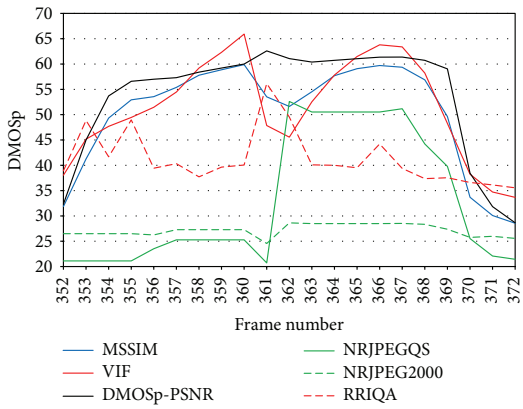


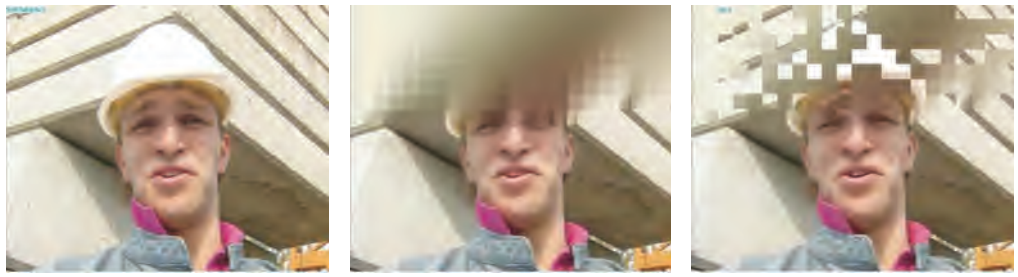
Figure 16: Detail from two consecutive long burst with incoming packets between them.

We explained the procedures that we followed to compare QAM metrics and alerted about some issues that arise when a comparison between heterogeneous metrics is made. The metrics must be compared using a common scale, since the raw scores of the metrics are not directly comparable. The scale-conversion process involves subjective tests and the use of mapping functions between the subjective MOS values and the metrics raw values. The parameters for the mapping function we used are provided in the paper. The metrics were first trained with a set of images from two open source image and video databases with known MOS values. The metrics were tested with another set of images and videos also taken from available databases. In order to perform a fair comparison, the training and testing sets used with each metric must use only impairments which the metric was designed to handle. We defined as heterogeneous metrics those that were trained with different sets of images or sequences. The R/D comparisons of heterogeneous metrics must be done carefully, focusing not only on the absolute quality scores, but also on the relative scoring between consecutive bitrates as the differences between DMOSp values are perceptually detected (or not) depending on the quality range. When metrics are trained with the same training set, differences in DMOSp values have the same perceptual meaning for all the metrics, but this may not be true between heterogeneous metrics. Normalizing the DMOSp scale when comparing heterogeneous metrics helps to detect these differences.

We performed the comparison between metrics in two environments: a compression environment and a packet loss environment. We performed several subjective tests in order to confirm that the analysis and the behavior of the metrics were consistent with human perception. Our tests included the comparisons of three encoders by replacing the PSNR as distortion metric in their R/D curves with each of the candidate metrics.

From our results of the compression environment, we conclude that we can trust the quality provided by the VIF metric, which is the one that obtains a better fit in terms of DMOS during the calibration process and on how it ranks the performance of the tested encoders for the bitrate range under consideration. The NRJPEG2000 and the RRIQA metrics break monotonicity for very high compression levels when M-JPEG2000 is the evaluated encoder. For the rest of the bitrates, all the other metrics show a monotonic behavior for all the bitrate range and for all encoders.

The choice of a QAM to replace the traditional PSNR, when working in a compression framework with no packet losses, depends on the availability of the reference sequence. In applications where the reference sequence is not available, RRIQA is our choice because its behavior is similar to FR metrics. If the reference sequence is available, the choice depends on the weight given to the tradeoff between computational cost and accuracy. If time is the most important parameter, we



(a) (b) (c)

Figure 17: Decoded frames between two consecutive bursts: (a) original frame; reconstructed frames (b) 361 and (c) 362.

will choose DMOSp-PSNR followed by VQM and MSSIM. If accuracy is more important, then the choice will be VIF and MSSIM metrics.

In the loss-prone environment, we analyzed the metrics behavior when measuring reconstructed video sequences encoded and delivered through error prone wireless networks, like MANETs. In order to obtain an accurate representation of delivery errors in MANETs, we adopted an HMM model able to represent different MANET scenarios.

The results of our analysis are as follows. (a) NR metrics are not able to properly detect and measure the sharp quality drop due to the loss of several consecutive frames. (b) The RR metric has a nondeterministic behavior in the presence of packet losses, having difficulties in identifying and measuring this effect when the video is encoded with moderate to high compression rates. (c) Concerning the other metrics, MSSIM, DMOSp-PSNR, and VIF show a similar behavior in all cases. In summary, we consider that although they exhibit slight differences in the packet loss framework, we propose the use of the MSSIM metric as a tradeoff between a high quality measurement process (resembling human visual perception) and computational cost.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

This research was supported by the Spanish Ministry of Education and Science under Grant no. TIN2011-27543-C03-03.S.

References

[1] K. Brunnstrom, D. Hands, F. Speranza, and A. Webster, "VQEG validation and ITU standardization of objective perceptual video quality metrics," *IEEE Signal Processing Magazine*, vol. 26, no. 3, pp. 96–101, 2009.

[2] J. Korhonen, N. Burini, J. You, and E. Nadernejad, "How to evaluate objective video quality metrics reliably," in *Proceedings of the 2012 4th International Workshop on Quality of Multimedia Experience (QoMEX '12)*, pp. 57–62, July 2012.

[3] M. P. Eckert and A. P. Bradley, "Perceptual quality metrics applied to still image compression," *Signal Processing*, vol. 70, no. 3, pp. 177–200, 1998.

[4] T. N. Pappas and R. J. Safranek, "Perceptual criteria for image quality evaluation," in *Handbook of Image and Video Processing*, pp. 669–684, Academic Press, 2000.

[5] VQEG, "Final report from the video quality experts group on the validation of objective models of video quality assessment," phase I, 2000.

[6] VQEG, "Final report from the video quality experts group on the validation of objective models of video quality assessment," phase II, 2003.

[7] Z. Wang, A. C. Bovik, and L. Lu, "Why is image quality assessment so difficult?" in *Proceedings of the 2002 IEEE International Conference on Acoustic, Speech, and Signal Processing (ICASSP '02)*, vol. 4, pp. 3313–3316, May 2002.

[8] 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, 2008.

[9] F. Porikli, A. Bovik, C. Plack et al., "Multimedia quality assessment [DSP Forum]," *IEEE Signal Processing Magazine*, vol. 28, no. 6, pp. 164–177, 2011.

[10] S. Winkler, "Issues in vision modeling for perceptual video quality assessment," *Signal Processing*, vol. 78, no. 2, pp. 231–252, 1999.

[11] Z. Wang, H. R. Sheikh, and A. C. Bovik, "Objective video quality assessment," in *The Handbook of Video Databases: Design and Applications*, chapter 41, pp. 1041–1078, CRC Press, 2003.

[12] B. Girod, "What's wrong with mean-squared error," in *Digital Images and Human Vision*, pp. 207–220, 1993.

[13] P. C. Teo and D. J. Heeger, "Perceptual image distortion," in *Proceedings of the IEEE International Conference on Image Processing (ICIP '94)*, vol. 2, pp. 982–986, 1994.

[14] C. J. van den Branden Lambrecht and O. Verscheure, "Perceptual quality measure using a spatiotemporal model of the human visual system," *Storage and Retrieval for Image and Video Databases*, vol. 2668, pp. 450–461, 1996.

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

[16] J. Malo, A. M. Pons, and J. M. Artigas, "Subjective image fidelity metric based on bit allocation of the human visual system in

the DCT domain," *Image and Vision Computing*, vol. 15, no. 7, pp. 535–548, 1997.

[17] A. B. Watson, "Toward a perceptual video-quality metric," in *Human Vision and Electronic Imaging III*, Proceedings of SPIE, July 1998.

[18] M. Masry and Y. Sermadevi, "A scalable wavelet-based video distortion metric and applications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 2, pp. 260–272, 2006.

[19] Z. Wang, A. C. Bovik, L. Lu, and J. Kouloheris, "Foveated wavelet image quality index," in *Applications for Digital Image Processing XXIV*, Proceedings SPIE, pp. 42–52, August 2001.

[20] A. Cavallaro and S. Winkler, "Segmentation-driven perceptual quality metrics," in *Proceedings of the 2004 International Conference on Image Processing (ICIP '04)*, vol. 5, pp. 3543–3546, October 2004.

[21] C. J. van den Branden Lambrecht, "A working spatio-temporal model of the human visual system for image restoration and quality assessment applications," in *Proceedings of the 996 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP' 96)*, vol. 4, pp. 2291–2294, May 1996.

[22] A. B. Watson, "The cortex transform: rapid computation of simulated neural images," *Computer Vision, Graphics, and Image Processing*, vol. 39, no. 3, pp. 311–327, 1987.

[23] J. Lubin, "The use of psychophysical data and models in the analysis of display system performance," in *Digital Images and Human Vision*, pp. 163–178, MIT Press, Cambridge, Mass, USA, 1993.

[24] S. Daly, "The visible differences predictor: an algorithm for the assessment of image fidelity," in *Digital Images and Human Vision*, pp. 179–206, MIT Press, Cambridge, Mass, USA, 1993.

[25] E. P. Simoncelli, W. T. Freeman, E. H. Adelson, and D. J. Heeger, "Shiftable multiscale transforms," *IEEE Transactions on Information Theory*, vol. 38, no. 2, pp. 587–607, 1992.

[26] S. Winkler, "Perceptual distortion metric for digital color video," in *Proceedings of the 1999 Human Vision and Electronic Imaging IV*, pp. 175–184, January 1999.

[27] A. B. Watson, "Dct quantization matrices visually optimized for individual images," 1993.

[28] M. Nadenau, *Integration of human color vision models into high quality image compression [Ph.D. thesis]*, STI, Lausanne, Switzerland, 2000.

[29] M. J. Nadenau, J. Reichel, and M. Kunt, "Performance comparison of masking models based on a new psychovisual test method with natural scenery stimuli," *Signal Processing: Image Communication*, vol. 17, no. 10, pp. 807–823, 2002.

[30] A. B. Watson and J. A. Solomon, "Model of visual contrast gain control and pattern masking," *Journal of the Optical Society of America A*, vol. 14, no. 9, pp. 2379–2391, 1997.

[31] C. Lambrecht and O. Verscheure, "Perceptual quality measure using a spatio-temporal model of the human visual system," vol. 2668 of *Proceedings of the SPIE*, pp. 450–461, San Jose, Calif, USA, January-February 1996.

[32] A. B. Watson, "Perceptual optimization of dct color quantization matrices," in *Proceedings of the 1994 IEEE International Conference on Image Processing (ICIP '94)*, vol. 1, pp. 100–104, 1994.

[33] S. Winkler, "Quality metric design: a closer look," in *Human Vision and Electronic Imaging*, vol. 3959 of *Proceedings of SPIE*, pp. 37–44, January 2000.

[34] A. B. Watson, G. Y. Yang, J. A. Solomon, and J. Villasenor, "Visibility of wavelet quantization noise," *IEEE Transactions on Image Processing*, vol. 6, no. 8, pp. 1164–1175, 1997.

[35] Z. Yu, H. R. Wu, S. Winkler, and T. Chen, "Vision-model-based impairment metric to evaluate blocking artifacts in digital video," *Proceedings of the IEEE*, vol. 90, no. 1, pp. 154–169, 2002.

[36] Y. Sermadevi and S. S. Hemami, "Linear programming optimization for video coding under multiple constraints," in *Proceedings of the Data Compression Conference (DCC '03)*, pp. 53–62, March 2003.

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

[38] S. Wolf and M. H. Pinson, "Spatial-temporal distortion metrics for in-service quality monitoring of any digital video system," in *Proceedings of the 1999 Multimedia Systems and Applications II*, pp. 266–277, September 1999.

[39] S. Wolf and M. Pinson, "Video quality measurement techniques," NTIA Technical Report TR-02-392, 2002.

[40] M. H. Pinson and S. Wolf, "A new standardized method for objectively measuring video quality," *IEEE Transactions on Broadcasting*, vol. 50, no. 3, pp. 312–322, 2004.

[41] S. Wolf and M. H. Pinson, "Low bandwidth reduced reference video quality moni- toring system," in *Proceedings of the 1st International Workshop on Video Processing and Quality Metrics for Consumer Electronics*, January 2005.

[42] S. Winkler, E. D. Gelasca, and T. Ebrahimi, "Perceptual quality assessment for video watermarking," in *Proceedings of the International Conference on Information Technology: Coding and Computing*, pp. 90–94, April 2002.

[43] Z. Wang, H. R. Sheikh, and A. C. Bovik, "No reference perceptual quality assessment of JPEG compressed images," in *Proceedings of the International Conference on Image Processing (ICIP '02)*, pp. 477–480, September 2002.

[44] P. Marziliano, F. Dufaux, S. Winkler, and T. Ebrahimi, "Perceptual blur and ringing metrics: application to JPEG2000," *Signal Processing: Image Communication*, vol. 19, no. 2, pp. 163–172, 2004.

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

[46] A. C. Bovik and S. Liu, "DCT-domain blind measurement of blocking artifacts in DCT-coded images," in *Proceedings of the 2001 IEEE Interntional Conference on Acoustics, Speech, and Signal Processing (ICASSP '01)*, vol. 3, pp. 1725–1728, May 2001.

[47] P. Marziliano, F. Dufaux, S. Winkler, and T. Ebrahimi, "A No-reference perceptual blur metric," in *Proceedings of the International Conference on Image Processing (ICIP'02)*, vol. 3, pp. 57–60, September 2002.

[48] T. M. Kusuma and H. J. Zepernick, "A reduced-reference perceptual quality metric for in-service image quality assessment," in *Proceedings of the Joint First Workshop on Mobile Future and Symposium on Trends in Communications (SympoTIC '03)*, pp. 71–74, 2003.

[49] 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, 2005.

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

[51] Z. Wang and A. C. Bovik, "A universal image quality index," *IEEE Signal Processing Letters*, vol. 9, no. 3, pp. 81–84, 2002.

[52] Z. Wang and E. P. Simoncelli, "An adaptive linear system framework for image distortion analysis," in *Proceedings of the IEEE International Conference on Image Processing 2005 (ICIP '05)*, vol. 3, pp. 1160–1163, September 2005.

[53] Z. Wang and E. P. Simoncelli, "Translation insensitive image similarity in complex wavelet domain," in *Proceedings of the 2005 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05)*, vol. 2, pp. 573–576, March 2005.

[54] Z. Wang, L. Lu, and A. C. Bovik, "Video quality assessment using structural distortion measurement," in *Proceedings of the International Conference on Image Processing (ICIP'02)*, vol. 3, pp. 65–68, September 2002.

[55] E. P. Simoncelli, "Modeling the joint statistics of images in the wavelet domain," in *44th Annual Meeting*, vol. 3813 of *Proceedings of SPIE*, pp. 188–195, July 1999.

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

[57] Z. Wang and E. P. Simoncelli, "Reduced-reference image quality assessment using a wavelet-domain natural image statistic model," in *Human Vision and Electronic Imaging X*, vol. 5666 of *Proceedings of SPIE*, pp. 149–159, January 2005.

[58] H. R. Sheikh, A. C. Bovik, and G. de Veciana, "An information fidelity criterion for image quality assessment using natural scene statistics," *IEEE Transactions on Image Processing*, vol. 14, no. 12, pp. 2117–2128, 2005.

[59] H. R. Sheikh and A. C. Bovik, "Image information and visual quality," *IEEE Transactions on Image Processing*, vol. 15, no. 2, pp. 430–444, 2006.

[60] VQEG, "Final report from the video quality experts group on the validation of objective models of video quality assessment," phase I, 2000.

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

[62] H. R. Sheikh, Z. Wang, L. Cormack, and A. C. Bovik, "Live image quality assessment database release 2," http://live.ece.utexas.edu/research/quality/.

[63] Video Quality Experts Group (VQEG), "Vqeg fr-tv phase i database," http://www.its.bldrdoc.gov/vqeg/downloads.aspx.

[64] A. M. Rohaly, P. J. Corriveau, J. M. Libert et al., "Video quality experts group: current results and future directions," in *Visual Communications and Image Processing*, K. N. Ngan, T. Sikora, and M. T. Sun, Eds., vol. 4067 of *Proceedings SPIE*, pp. 742–753, May 2000.

[65] "Coding of audiovisual objects part 10: advanced videocoding," ISO/IEC 14496-10:2003, ITUT Recommendation H264 Advanced video codingfor generic audiovisual services, 2003.

[66] "JPEG 2000 image coding system, part 1: core coding system," ISO/IEC 15444-1, 2000.

[67] J. Oliver and M. P. Malumbres, "Low-complexity multiresolution image compression using wavelet lower trees," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 11, pp. 1437–1444, 2006.

[68] C. T. Calafate, P. Manzoni, and M. P. Malumbres, "Speeding up the evaluation of multimedia streaming applications in MANETs using HMMs," in *Proceedings of the 7th ACM Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (ACM MSWiM '04)*, pp. 315–322, October 2004.

[69] "Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements," IEEE 802.11 WG. 802.11e, 2005.

[70] C. T. Calafate, M. P. Malumbres, and P. Manzoni, "Performance of H.264 compressed video streams over 802.11b based MANETs," in *Proceedingsof the 24th International Conference on Distributed Computing Systems Workshops (ICDCSW '04)*, vol. 7, pp. 776–781, March 2004.

Martínez-Rach *et al. EURASIP Journal on Advances in Signal Processing* 2013, **2013**:141
http://asp.eurasipjournals.com/content/2013/1/141

EURASIP Journal on Advances in Signal Processing
a SpringerOpen Journal

RESEARCH Open Access

Enhancing LTW image encoder with perceptual coding and GPU-optimized 2D-DWT transform

Miguel O Martínez-Rach*, Otoniel López-Granado, Vicente Galiano, Hector Migallón, Jesús Llor and Manuel P Malumbres

Abstract

When optimizing a wavelet image coder, the two main targets are to (1) improve its rate-distortion (R/D) performance and (2) reduce the coding times. In general, the encoding engine is mainly responsible for achieving R/D performance. It is usually more complex than the decoding part. A large number of works about R/D or complexity optimizations can be found, but only a few tackle the problem of increasing R/D performance while reducing the computational cost at the same time, like Kakadu, an optimized version of JPEG2000. In this work we propose an optimization of the E_LTW encoder with the aim to increase its R/D performance through perceptual encoding techniques and reduce the encoding time by means of a graphics processing unit-optimized version of the two-dimensional discrete wavelet transform. The results show that in both performance dimensions, our enhanced encoder achieves good results compared with Kakadu and SPIHT encoders, achieving speedups of 6 times with respect to the original E_LTW encoder.

Keywords: Wavelet image coding; Perceptual coding; Contrast sensitivity function; GPU optimization

1 Introduction

Wavelet transforms have been reported to have good performance for image compression; therefore, many state-of-the-art image codecs, including the JPEG2000 image coding standard, use the discrete wavelet transform (DWT) [1,2]. The use of wavelet coefficient trees and successive approximations was introduced by the embedded zerotree wavelet (EZW) algorithm [3] with a bitplane coding approximation. SPIHT [2], an advanced version of EZW, processes the wavelet coefficient trees in a more efficient way by partitioning the coefficients depending on their significance. Both EZW and SPIHT need the coefficient tree construction to search for significant coefficients through a multiple iterative process at each bitplane, which involves high computational complexity.

Bitplane coding is implemented by the JPEG2000 encoding codeblocks with three passes per plane, so the most important information, from a rate-distortion (R/D) point of view, is first encoded. It also uses an optional and low-complexity post-compression optimization algorithm, based on the Lagrange multiplier method. Besides, it uses a large number of contexts for the arithmetic encoder. This post-compression rate-distortion optimization algorithm selects the most important coefficients by weighting them, based on the mean square error (MSE) distortion measurement.

Wavelet-based image processing systems are typically implemented with memory-intensive algorithms and with higher execution time than other encoders based on other transforms like discrete cosine transform. In usual two-dimensional (2D)-DWT implementations [4], image decomposition is computed by means of a convolution filtering process, and so its complexity rises as the filter length does. The image is transformed at every decomposition level, first column by column and then row by row.

In [5], the authors proposed the E_LTW codec with sign coding, precise rate-control, and some optimizations to avoid bitplane processing, at the cost of not being embedded, but with low memory requirements and similar R/D performance than the one obtained by embedded encoders like JPEG2000 and SPIHT.

*Correspondence: mmrach@umh.es
Physics and Computer Architecture Department, Miguel Hernández University, Elche 03202, Spain

Part II of the JPEG2000 standard includes visual progressive weighting [6] and visual masking by setting the weights based on the human visual system (HVS) using contrast sensitivity function (CSF). Many other image encoders have included much of the knowledge of the human visual system in order to obtain a better perceptual quality of the compressed images. The most widely used characteristic is the contrast adaptability of the HVS, because HVS is more sensitive to contrast than to absolute luminance [7]. The CSF relates the spatial frequency with the contrast sensitivity.

This perceptual coding will improve the perceptual quality of the reconstructed images, so that for a desired rate range, a better perceptual R/D behavior is achieved.

Although most studies employ the peak signal-to-noise ratio (PSNR) metric to measure image quality performance, it is well known that this metric does not always capture the distortion perceived by the human being. Therefore, we decided to use objective quality assessment metrics whose design is inspired by the HVS, since our proposal includes perceptual-based encoding techniques that may not be properly evaluated by the PSNR metric.

In this work, we propose the PE_LTW (perceptually enhanced LTW) as an enhanced version of the E_LTW encoder by including perceptual coding based on the CSF and the use of graphics processing unit (GPU)-optimized 2D-DWT algorithms based on the methods described in [4,8].

After improving the perceptual R/D behavior of our proposal, we proceed to optimize the 2D-DWT transform module by GPU processing to reduce the overall encoding time. From previous work, we have defined a CUDA implementation of the 2D-DWT transform that is able to considerably reduce the 2D-DWT computation time.

So as to test the behavior of our proposal, we have compared the performance of our PE_LTW encoder in terms of perceptual quality and encoding delays with the Kakadu implementation of the JPEG2000 standard, with and without enabling its perceptual weighting mode, and with the SPIHT image encoder.

2 Encoding system

2.1 Encoder

The basic idea of this encoder is very simple: after computing the 2D-DWT transform of an image, the perceptually weighted wavelet coefficients are uniformly quantized and then encoded with arithmetic coding.

As mentioned, the 2D-DWT computation stage runs on a GPU and includes the perceptual weighting based on the CSF and implemented as an invariant scaling factor weighting (ISFW) [9] that weights the obtained coefficients depending on the importance that the frequency subband has for the HVS contrast sensitivity. We detail the CSF and the ISFW later in the next sections.

The uniform quantization of the perceptually weighted coefficients is performed by means of two strategies: one coarser and another finer. The finer one consists of applying a scalar uniform quantization (Q) to the coefficients. The coarser one is based on removing the least significant bitplanes (*rplanes*) from coefficients.

For the coding stage, if the absolute value of a coefficient and all its descendants (considering the classic quad-tree structure from [2]) is lower than a threshold value ($2^{rplanes}$), the entire tree is encoded with a single symbol, which we call LOWER symbol (indicating that all the coefficients in the tree are lower than $2^{rplanes}$ and so they form a lower tree). However, if a coefficient is lower than the threshold and not all its descendants are lower than it, that coefficient is encoded with an ISOLATED LOWER symbol. On the other hand, for each wavelet coefficient higher than $2^{rplanes}$, we encode a symbol indicating the number of bits needed to represent that coefficient, along with a binary-coded representation of its bits and sign (note that the *rplanes* less significant bits are not encoded).

The encoder exploits the sign neighborhood correlation of wavelet subband type (HL,LH,HH) as Deever and Hemami assessed in [10] by encoding the prediction of the sign (success of failure).

The proposed encoder also includes the rate control algorithm presented in [11] but taking into account the sign coding and the intrinsic error model of the rate control. As the rate control underestimates the target rate, the required bits to match the target bitrate are added to the bitstream. The selected bits correspond to the bitplanes (lower or equal to the *rplanes* quantization parameter) of significant coefficients added to the output bitstream following a particular order, from low-frequency subbands to the highest one.

More details about the coding and decoding algorithms, along with a formal description and an example of use, can be found in [5,12].

2.2 The contrast sensitivity function

In [9], the authors explained how the sensitivity to contrast of the HVS can be exploited by means of the CSF curve to enhance the perceptual or subjective quality of the DWT-encoded images. A comprehensive review of HVS models for quality assessment/image compression is found in [7]. Most of these models take into account the varying sensitivity over spatial frequency, color, and the inhibiting effects of strong local contrasts or activity, called masking.

Complex HVS models implement each of these low-level visual effects as a separate stage. Then the overall model consists of the successive processing of each stage. One of the initial HVS stages is the visual sensitivity as a function of spatial frequency that is described by the CSF.

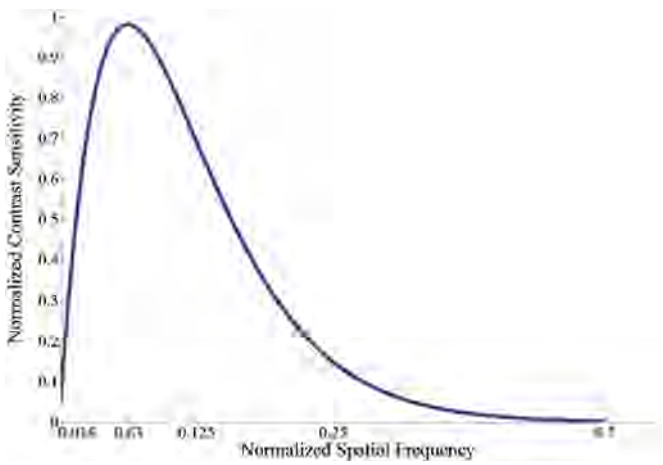


Figure 1 Contrast sensitivity function.

A closed-form model of the CSF for luminance images [13] is given by

$$H(f) = 2.6(0.0192 + 0.114f)e^{-(0.114f)^{1.1}} \quad (1)$$

where spatial frequency is $f = (f_x^2 + f_y^2)^{1/2}$ with units of cycles/degree (f_x and f_y are the horizontal and vertical spatial frequencies, respectively). The frequency is usually measured in cycles per optical degree, which makes the CSF independent of the viewing distance.

Figure 1 depicts the CSF curve obtained with Equation 1, and it characterizes luminance sensitivity as a function of normalized spatial frequency (CSF = 1/Contrast threshold). As shown, CSF is a band-pass filter, which is most sensitive to normalized spatial frequencies between 0.025 and 0.125 and less sensitive to very low and very high frequencies. The reason why we cannot distinguish patterns with high frequencies is the limited number of photoreceptors in our eyes. CSF curves exist for chrominance as well. However, unlike luminance stimuli, human sensitivity to chrominance stimuli is relatively uniform across spatial frequency.

One of the first works that demonstrate that the MSE cannot reliably predict the difference of the perceived quality of two images can be found in [13]. They propose, by way of psychovisual experiments, the aforementioned model of the CSF, which is well suited and widely used [6,14-16] for wavelet-based codecs; therefore, we adopt this model.

2.3 Using the CSF

In [9], the authors explained how the CSF can be implemented in wavelet-based codecs. Some codecs, like the JPEG2000 standard Part II, introduce the CSF as a visual progressive single factor weighting, replacing the MSE by the CSF-weighted MSE (WMSE) and optimizing system parameters to minimize WMSE for a given bitrate. This is done in the post-compression rate-distortion optimization algorithm where the WMSE replaces the MSE as the cost function which drives the formation of quality layers [6].

CSF weights can be obtained also by applying to each frequency subband the appropriate contrast detection threshold. In [15], subjective experiments were performed to obtain a model that expresses the threshold DWT noise as a function of spatial frequency. Using this model, the authors obtained a perceptually lossless quantization matrix for the linear phase 9/7 DWT. By the use of this quantization matrix, each subband is quantized by a value that weights the overall resulting quantized image at the threshold of artifacts visibility. For suprathreshold quantization, a uniform quantization stage is afterward performed.

However, we introduce the CSF in the encoder using the ISFW strategy proposed also in [9]. So from the CSF curve, we obtain the weights for scaling the wavelet coefficients. This weighting can be introduced after the wavelet filtering stage and before the uniform quantization stage

Table 1 Proposed CSF weighting matrix

	LL	LH	HH	HL
L1	1.0	1.1795	1.0000	1.7873
L2	1.0	3.4678	2.4457	4.8524
L3	1.0	6.2038	5.5841	6.4957
L4	1.0	6.4177	6.4964	6.1187
L5	1.0	5.1014	5.5254	4.5678
L6	1.0	3.5546	3.9300	3.1580

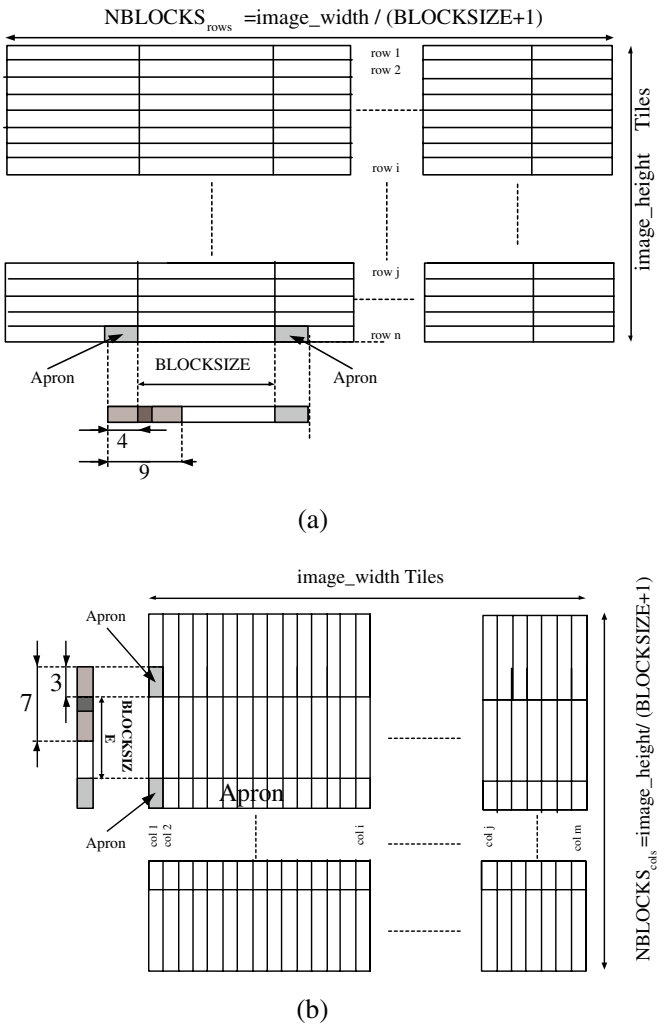


Figure 2 Shared memory for the Daubechies 9/7 filter. (a) Shared memory for the row filter. **(b)** Shared memory for the column filter.

is applied. The weighting is a simple multiplication of the wavelet coefficients in each frequency subband by the corresponding weight. At the decoder, the inverse of this weight is applied. The CSF weights do not need to be explicitly transmitted to the decoder. This stage is independent to the other encoder modules (wavelet filtering, quantization, etc).

The granularity of the correspondence between frequency and weighting value is a key issue. As wavelet-based codecs obtain a multiresolution signal decomposition, the easiest association is to find a unique weighting value (or contrast detection threshold) for each wavelet frequency subband. If further decompositions of the frequency domain are done, for example, a finer association could be done between frequency and weights using packet wavelets [17].

We perform the ISFW implementation based on [18] but increasing the granularity at the subband level. This is done in the wavelet transform stage of the PE-LTW encoder multiplying each coefficient in a wavelet subband by its corresponding weighting factor. In spite of the fact that CSF (Equation 1) is independent of the viewing distance, in order to introduce it as a scaling factor, the resolution and the viewing distance must be fixed. Although an observer can look at the images from any distance, as stated in [9], the assumption of 'worst case

Table 2 GPU vs. SEQ PE_LTW speedup and total encoding time comparison with SPIHT and Kakadu

	Rates	PE_LTW					SPIHT	Kakadu
	(bpp)	SEQ-DWT	GPU-DWT	Rate & Coder	T.SEQ	T.GPU	Speedup	Total	Total
Lena	1.00	17.08	0.85	31.80	48.88	32.65	1.50	93.04	13.00
	0.5	17.23	0.86	16.15	33.38	17.01	1.96	185.74	9.00
	0.25	17.17	0.86	10.39	27.56	11.25	2.45	198.64	8.00
	0.125	17.57	0.88	7.73	25.30	8.61	2.94	220.15	7.00
Barbara	1.00	17.89	0.89	27.26	45.15	28.16	1.60	77.80	15.00
	0.5	17.42	0.87	17.04	34.46	17.91	1.92	72.37	9.00
	0.25	17.45	0.87	11.53	28.98	12.40	2.34	42.59	8.00
	0.125	17.49	0.87	8.38	25.87	9.25	2.79	35.04	7.00
Goldhill	1.00	17.61	0.88	30.62	48.23	31.50	1.53	99.46	12.00
	0.5	18.13	0.91	18.21	36.34	19.12	1.90	52.72	24.00
	0.25	17.30	0.86	11.51	28.81	12.38	2.33	45.51	8.00
	0.125	17.42	0.87	7.97	25.39	8.84	2.87	28.86	7.00
Boat	1.00	17.02	0.85	27.44	44.46	28.29	1.57	79.05	11.00
	0.5	17.35	0.87	17.13	34.49	18.00	1.92	51.22	9.00
	0.25	17.03	0.85	11.35	28.37	12.20	2.33	41.98	7.00
	0.125	17.13	0.86	7.95	25.07	8.80	2.85	59.12	8.00
Mandrill	1.00	17.99	0.90	32.85	50.84	33.75	1.51	94.06	19.00
	0.5	17.89	0.89	19.98	37.87	20.87	1.81	51.86	11.00
	0.25	17.59	0.88	13.11	30.69	13.99	2.19	40.83	8.00
	0.125	17.87	0.89	8.59	26.46	9.48	2.79	47.26	8.00
Balloon	1.00	16.89	0.84	26.86	43.75	27.71	1.58	104.25	12.00
	0.5	17.27	0.86	16.39	33.67	17.26	1.95	45.25	9.00
	0.25	16.89	0.84	10.92	27.81	11.77	2.36	36.91	8.00
	0.125	16.89	0.84	8.06	24.95	8.90	2.80	29.03	7.00
Horse	1.00	17.60	0.88	31.81	49.42	32.69	1.51	86.45	13.00
	0.5	17.34	0.87	18.49	35.83	19.36	1.85	56.35	9.00
	0.25	17.33	0.87	11.38	28.71	12.25	2.34	36.74	9.00
	0.125	17.55	0.88	8.25	25.80	9.12	2.83	43.10	8.00
Zelda	1.00	17.11	0.86	35.36	52.48	36.22	1.45	57.56	11.00
	0.5	17.08	0.85	16.58	33.65	17.43	1.93	34.68	9.00
	0.25	17.39	0.87	10.48	27.87	11.35	2.46	25.36	8.00
	0.125	17.25	0.86	7.40	24.65	8.26	2.98	26.44	7.00
Cafe	1.00	419.10	20.95	521.75	940.85	542.71	1.73	719.54	197.00
	0.5	418.50	20.92	325.41	743.91	346.34	2.15	1,854.99	129.00
	0.25	418.97	20.95	217.20	636.17	238.15	2.67	1,104.76	105.00
	0.125	418.73	20.94	150.93	569.66	171.86	3.31	733.09	90.00
Bike	1.00	412.87	20.64	508.61	921.48	529.26	1.74	1265.46	171.00
	0.5	413.13	20.66	296.34	709.47	317.00	2.24	1867.98	121.00
	0.25	415.15	20.76	191.44	606.59	212.20	2.86	943.82	101.00
	0.125	414.18	20.71	134.58	548.76	155.29	3.53	762.22	88.00
Woman	1.00	414.49	20.72	527.83	942.31	548.55	1.72	819.65	169.00
	0.5	414.12	20.71	321.25	735.36	341.95	2.15	1,528.94	137.00
	0.25	418.81	20.94	215.76	634.57	236.70	2.68	913.84	95.00
	0.125	417.78	20.89	151.65	569.43	172.54	3.30	699.80	89.00

viewing conditions' can produce CSF weighting factors that work properly for all different viewing distances and media resolutions. So after fixing viewing conditions, we obtain the weighting matrix, presented in Table 1. For each wavelet decomposition level and frequency orientation, the weights are directly obtained from the CSF curve, by normalizing the corresponding values so that the most perceptually important frequencies are scaled with higher values, while the less important are preserved. This scaling process augments the magnitude of all wavelet coefficients, except for those in the LL subband that are neither scaled nor quantized in our coding algorithm. Our tests reveal that, thanks to the weighting process, the uniform quantization stage preserves a very good balance between bitrate and perceptual quality in all the quantization range, from under-threshold (perceptually lossless) to suprathreshold quantization (lossy).

2.4 GPU 2D-DWT optimization

In order to develop the 2D-DWT-optimized version, we will use an NVIDIA GTX 280 GPU that contains 30 multiprocessors with eight cores in each multiprocessor, 1 GB of global memory, and 16 kB of shared memory (SM) by block.

Firstly, we will define our GPU-based 2D-DWT algorithm, named as CUDA Conv 9/7, as the reference algorithm. It will only use the GPU shared memory space to store the buffer that will contain a copy of the working row/column data. The constant memory space is used to store the filter taps. We call each CUDA kernel with a one-dimensional number of thread blocks, NBLOCKS, and a one-dimensional number of threads by block, NTHREADS.

In the horizontal DWT filtering process, each image row is stored in the threads shared memory. After that, in the vertical filtering, each column is processed in the same way. The row or column size determines the NBLOCKS parameter, which must be greater or equal to the image width in the horizontal step or the image height in the vertical step. One of the goals in the proposed CUDA-based methods is not to increase memory requirements, so we will store the resulting wavelet coefficients in the original image memory space.

For computing the DWT, the threads use the shared memory space, where latency access is extremely low. The CUDA-Sep 9/7 algorithm stores the original image in the GPU global memory but computes the filtering steps from the shared memory.

Execution in the GPU is composed by threads grouped in a number of 32 threads called warp. Each warp must load a block of the image from the global memory into a shared memory array with BLOCKSIZE pixels. As it can be seen in Figure 2, the number of thread blocks, NBLOCKS, or tiles depends on BLOCKSIZE and image dimensions. Moreover, pixels located in the border of the block also need neighbor pixels from other blocks to compute the convolution. These regions are called apron and are shadowed in the last row and column of Figure 2a, b. The size of the apron region depends on the filter radius (the filter radius being the half of the filter length minus 1). In both figure panels, the values of the filter radius and the filter length corresponding to the Daubechies 9/7 filter are presented.

We can reduce the number of idle threads by reducing the total number of threads per block and also using each thread to load multiple pixels into the shared memory. This ensures that all threads of each warp are active during the computation stage. Note that the number of threads in a block must be a multiple of the warp size (32 threads on GTX 280) for optimal efficiency.

To achieve higher efficiency and higher memory throughput, the GPU attempts to coalesce accesses from multiple threads into a single memory transaction. If all threads within a warp (32 threads) simultaneously read consecutive words, then a single large read of the 32 values can be performed at optimum speed. In the CUDA-Sep

Table 3 Speedup comparison by target bitrate

	Rates	PE_LTW mean times	SPIHT	Kakadu	Speedup comparison	
	(bpp)	T.GPU	Total	Total	vs. SPIHT	vs. Kakadu
512 × 512	1	31.4	86.5	13.3	2.76	0.42
	0.5	18.4	68.8	11.1	3.74	0.61
	0.25	12.2	58.6	8.0	4.80	0.66
	0.125	8.9	61.1	7.4	6.86	0.83
2,048 × 2,560	1	540.2	934.9	179.0	1.73	0.33
	0.5	335.1	1,750.6	129.0	5.22	0.38
	0.25	229.0	987.5	100.3	4.31	0.44
	0.125	166.6	731.7	89.0	4.39	0.53

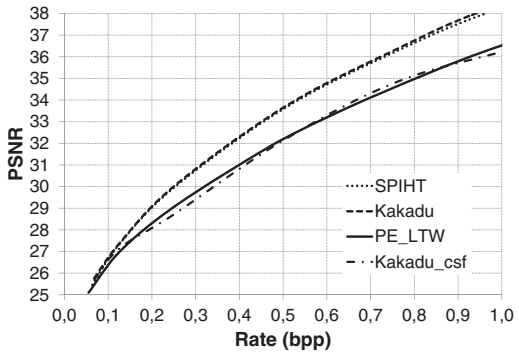


Figure 3 PSNR R/D comparison of the Woman image encoded with PE_LTW, SPIHT, and Kakadu. Rates are in bits per pixel.

9/7 algorithm, the convolution process is separated in two stages:

1. The row filtering stage
2. The column filtering stage

Each row/column filtering stage is separated into two substages: (a) the threads load a block of pixels of one row/column from the global memory into the shared memory, and (b) each thread computes the filter over the data stored in the shared memory and the result is sent to the global memory. For the column filtering, the resulting coefficient is stored in the global memory after performing the perceptual weighting, i.e., multiplying the final coefficient by the perceptual weight corresponding to the wavelet subband of the coefficient.

In the row or column filtering, the pixels located in the image block borders also need adjacent pixels from other thread blocks to compute the DWT. The apron region must also be loaded in the shared memory, but only for reading purposes, because the filtered value of the pixels located there is computed by other thread blocks.

The speedup achieved by the DWT GPU-based algorithm is up to 20 times relative to the sequential implementation in one core. Note that wavelet transform is only a single first step in an image/video encoder.

3 Performance evaluation

All evaluated encoders have been tested on an Intel Pentium Core 2 CPU at 1.8 GHz with 6 GB of RAM memory. We use an NVIDIA GTX 280 GPU that contains 30 multiprocessors with eight cores in each multiprocessor, 1 GB of global memory, and 16 kB of shared memory by block (or SM).

The proposed encoder is compared with Kakadu 5.2.5 and SPIHT (Sphit 8.01) encoders with two sets of test images: (a) a 512 × 512 image resolution set including

Figure 4 Subjective comparison of the Woman image encoded at 0.25 bpp. (a) SPIHT (PSNR = 29.95 dB). **(b)** Kakadu (PSNR = 30.01 dB). **(c)** PE_LTW (PSNR = 29.11 dB).

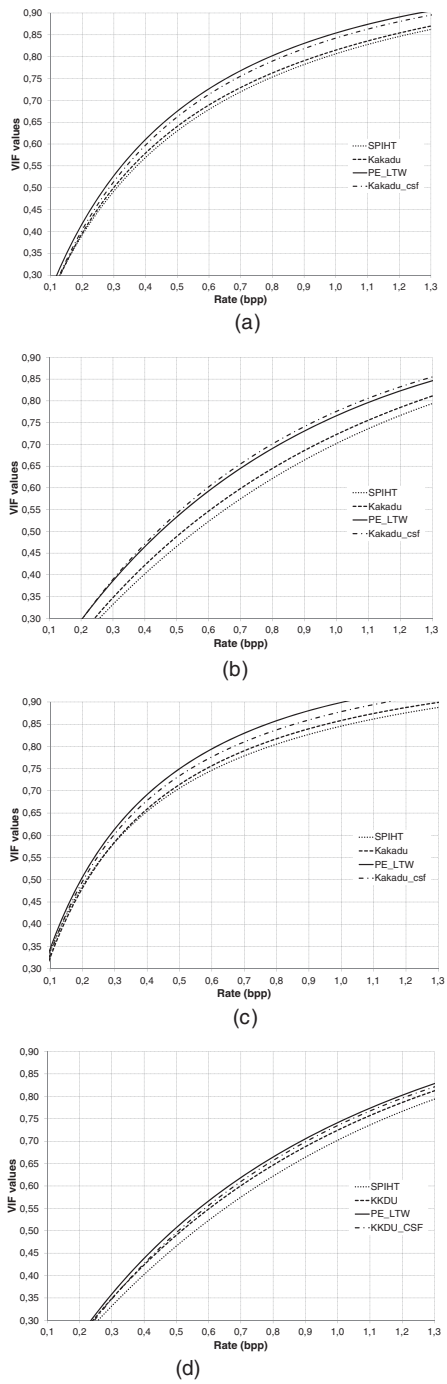


Figure 5 VIF R/D comparisons for different images from the test set. (a) Lena. **(b)** Barbara. **(c)** Zelda. **(d)** Woman.

Lena, Barbara, Balloon, Horse, Goldhill, Boat, Mandrill, and Zelda, and (b) a $2,048 \times 2,560$ image resolution set including Cafe, Bike, and Woman. When comparing with Kakadu, we perform two comparisons: one labeled as Kakadu_csf, which has enabled its perceptual weighting mode (with the perceptual weights presented in [6]), and the other one, labeled as Kakadu, without perceptual weights.

First, we analyze the speedup of the GPU-based encoder using 2D-DWT described in the previous section with respect to the traditional convolution algorithm running in a single core processor.

In Table 2, we show for each test image, at different bitrates, the encoding times for SPIHT, Kakadu, and our proposal in milliseconds. The first six columns are related to our proposal: The *SEQ-DWT* column shows the time required by the DWT when running on a single core. The *GPU-DWT* column shows the time of the CUDA-Sep 9/7 DWT version when running on GPU. The *Rate & Coder* column shows the time required by the rate control and the encoding stage, this time being common for both the sequential and GPU 2D-DWT versions. The *T.SEQ* column shows the total time for the sequential version and the *T.GPU* the total time for the GPU version. Finally, the *Speedup* column shows the speedup of the GPU version compared to the sequential version. The last two columns are the total execution time, also in milliseconds, for the other encoders, SPIHT and Kakadu.

When the target bitrate is low, i.e., high compression rate, the uniform quantization of the wavelet coefficients produces a great number of nonsignificant coefficients in low decomposition levels, the root of the zero tree being located at higher decomposition levels. This fact reduces the computation cost because only the root of a zero tree needs to be encoded. As a consequence, the overall number of operations is reduced and the gain of GPU optimized version is reduced too.

Table 3 shows the comparison of the average execution times (milliseconds) of each image in the test set at different compression rates. The PE_LTW is faster than SPIHT regardless of the target rate for any image size. However, the Kakadu encoder is still faster than the PE_LTW. Although the PE_LTW runs its DWT stage over the GPU, it is the only optimized stage in the whole encoder. By contrast, all encoding stages in the Kakadu 5.2.5 are fully optimized. Besides the use of multithread and multicore hardware capabilities, Kakadu uses processor intrinsics capabilities like MMX/SSE/SSE2/SIMD and uses a very fast multicomponent transform, i.e., block transform, which is well suited for parallelization.

4 R/D evaluation

For evaluating image encoders, the most common performance metric is the well-known R/D, the trade-off

between encoder bitrate (bpp) and the reconstructed quality typically measured in decibels through the PSNR of luminance color plane. However, it is also well known that the PSNR quality measurement is not close to the human perception of quality and sometimes it gives wrong quality scores, leading to erroneous conclusions when evaluating different encoding strategies.

Figure 3 shows the R/D comparison of the Woman ($2,048 \times 2,560$) image compressed with the PE_LTW encoder, SPIHT, Kakadu, and Kadadu_csf, using PSNR as quality metric. A misleading conclusion after looking at R/D curves for the PE_LTW and Kakadu_csf is that the encoding strategy of those proposals are inappropriate, since their quality results are always lower than those of the other encoders, specially at high bitrates.

There are several studies about the convenience of using other image quality assessment metrics than PSNR that better fit to human perceptual quality assessment (i.e., subjective test results) [14,17,19,20]. One of the best behaving objective quality assessment metrics is visual information fidelity (VIF) [7], which has been proven [17,19] to have a better correlation with subjective perception than other metrics that are commonly used for encoder comparisons [14,20]. The VIF metric uses statistic models of natural scenes in conjunction with distortion models in order to quantify the statistical information shared between the test and reference images.

As an example of how measuring the perceptual quality of images with PSNR is misleading, we show in Figure 4 a subjective comparison of the three encoders with a cropped region of the Woman test image compressed at 0.25 bpp. In this case the third image, encoded with PE_LTW, seems to have better subjective quality than the other two. This observation contradicts the conclusion obtained from Figure 3 that suggests that at this rate PE_LTW is worse than SPIHT and Kakadu. The same behavior can be observed as well with the other test images. So it is better not to trust on how PNSR ranks quality and use instead a perceptually inspired quality assessment metric like VIF that, as stated in [17,19], has a better correlation with the human perception of image quality.

So we will use the VIF metric in our R/D comparisons. Figure 5 shows some of the R/D results for some test images. As shown, the PE_LTW encoder can achieve higher compression rates while maintaining the same perceptual quality than the other encoders, i.e., a bitrate saving is obtained while using the PE_LTW instead of Kakadu or SPIHT at a desired quality.

Table 4 shows the rate savings obtained with PE_LTW vs. Kakadu, SPIHT, and Kakadu_csf. The VIF interval varies from 0.1 to 0.95 VIF quality units, 0.1 being the worst quality. This table groups the results by image resolution. Results are expressed as percentages of saved rate in the aforementioned VIF interval.

5 Conclusions

We have presented a perceptual image wavelet encoder whose 2D-DWT stage is implemented using CUDA running on a GPU. Our proposed perceptual encoder reveals the importance of exploiting the contrast sensitivity function behavior of the HVS by means of an accurate

Table 4 Rate savings of PE_LTW vs. Kakadu, SPIHT, and Kakadu with perceptual weights *Kakadu_csf*

P_ELTW images	vs. Kakadu (% rate saved, mean)	vs. SPIHT (% rate saved, mean)	vs. Kakadu_csf (% rate saved, mean)
512 × 512			
Lena	13.87	16.83	5.23
Barbara	11.39	17.44	−2.61
Goldhill	7.76	13.07	0.09
Boat	8.58	12.02	0.47
Mandrill	19.13	22.01	3.08
Balloon	10.45	10.75	2.16
Horse	14.96	14.91	3.74
Zelda	17.22	20.43	8.46
Mean 512 × 512	12.92	15.93	2.58
2,048 × 2,560			
Cafe	9.63	12.34	1.43
Bike	9.24	15.57	−0.80
Woman	5.21	11.46	3.75
Mean 2,048 × 2,560	8.03	13.12	1.46

perceptual weighting of wavelet coefficients. PE_LTW is very competitive in terms of perceptual quality, being able to obtain important bitrate savings regardless of the image resolution and at any bitrate when compared with SPIHT and Kakadu with and without its perceptual weighting mode enabled. The PE_LTW encoder is able to produce a quality-equivalent image with respect to the other two encoders with a reduced rate.

As the 2D-DWT transform runs on a GPU, the overall encoding time is highly reduced compared to the sequential version of the same encoder, obtaining maximum speedups of 6.86 for 512×512 images and 4.39 for $2,048 \times 2,560$ images. Compared with SPIHT and Kakadu, our proposal is clearly faster than SPIHT but needs additional optimizations to outperform Kakadu times.

Competing interests
The authors declare that they have no competing interests.

Acknowledgements
This research was supported by the Spanish Ministry of Education and Science under grant TIN2011-27543-C03-03.S

Received: 31 January 2013 Accepted: 1 August 2013
Published: 23 August 2013

References

1. ISO, ISO/IEC JTC 1/SC 29/WG 1 N1890. JPEG 2000 image coding system. Part 1: core coding system (ISO Geneva, 2000)
2. A Said, A Pearlman, A new, fast and efficient image codec based on set partitioning in hierarchicaltrees. IEEE Trans. Circ., Syst. Video Technol. **6**(3), 243–250 (1996)
3. JM Shapiro, A fast technique for identifying zerotrees in the EZW algorithm. Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. **3**, 1455–1458 (1996)
4. SG Mallat, A theory for multi-resolution signal decomposition: the wavelet representation. IEEE Trans. Pat. Anal. Mach. Intel. **11**(7), 674–693 (1989)
5. O Lopez, M Martinez, P Pinol, MP Malumbres, J Oliver, in *2009 16th IEEE International Conference on Image Processing (ICIP)*. E-LTW: an enhanced LTW encoder with sign coding and precise rate control (IEEE Piscataway, 2009), pp. 2821–2824
6. DS Taubman, MW Marcellin, *JPEG2000 Image Compression Fundamentals, Standards and Practice*. (Springer, Berlin, 2002)
7. HR Sheikh, G AC Bovik, de Veciana, An information fidelity criterion for image quality assessment using natural scene statistics. IEEE Trans. Image Process. **14**(12), 2117–2128 (2005)
8. W Sweldens, The lifting scheme: a custom-design construction of biorthogonal wavelets. Appl. Comput. Harmonic Anal. **3**(2), 186–200 (1996)
9. MJ Nadenau, J Reichel, M Kunt, Wavelet-based color image compression: exploiting the contrast sensitivity function. IEEE Trans. Image Process. **12**(1), 58–70 (2003)
10. A Deever, SS Hemami, in *Proceedings of the Data Compression Conference, 2000 (DCC 2000)*. What's your sign?: efficient sign coding for embedded wavelet image coding (IEEE, 2000), pp. 273–282
11. O López, M Martinez-Rach, J Oliver, MP Malumbres, in *Visual Communications and Image Processing 2007*. Impact of rate control tools on very fast non-embedded wavelet image encoders (IEEE Piscataway, 2007)
12. J Oliver, MP Malumbres, Low-complexity multiresolution image compression using wavelet lower trees. IEEE Trans. Circ. Syst. Video Technol. **16**(11), 1437–1444 (2006)
13. J Mannos, D Sakrison, The effects of a visual fidelity criterion of the encoding of images. IEEE Trans. Info. Theory. **20**(4), 525–536 (1974)
14. Z Wang, A Bovik, H Sheikh, EP Simoncelli, Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. **13**(4), 600–612 (2004)
15. AB Watson, GY Yang, JA Solomon, J Villasenor, Visibility of wavelet quantization noise. IEEE Trans. Image Process. **6**(8), 1164–1175 (1997)
16. N Moumkine, A Tamtaoui, A Ait Ouahman, in *Proceedings of the Second International Symposium on Communications, Control and Signal Processing (ISCCSP 2006)*. Integration of the contrast sensitivity function into wavelet codec (Marrakech, 13–15 Mar 2006)
17. X Gao, W Lu, D Tao, X Li, Image quality assessment based on multiscale geometric analysis. IEEE Trans. Image Process. **18**(7), 1409–1423 (2009)
18. AP Beegan, LR Iyer, AE Bell, VR Maher, MA Ross, in *Proceedings of the 2002 IEEE 10th Digital Signal Processing Workshop, 2002 and the 2nd Signal Processing Education Workshop*, vol. 2002. Design and evaluation of perceptual masks for wavelet image compression (IEEE Piscataway, pp. 88–93
19. M Martinez-Rach, O Lopez, P Piñol, J Oliver, MP Malumbres, in *Eight IEEE International Symposium on Multimedia*, vol.1. A study of objective quality assessment metrics for video codec design and evaluation IEEE Computer Society San Diego, 2006), pp. 517–524
20. HR Sheikh, MF Sabir, AC Bovik, A statistical evaluation of recent full reference image quality assessment algorithms. IEEE Trans. Image Process. **15**(11), 3440–3451 (2006)

doi:10.1186/1687-6180-2013-141
Cite this article as: Martínez-Rach *et al.*: Enhancing LTW image encoder with perceptual coding and GPU-optimized 2D-DWT transform. *EURASIP Journal on Advances in Signal Processing* 2013 **2013**:141.

Bibliography

[1] Zhou Wang and A.C. Bovik. Mean squared error: Love it or leave it? a new look at signal fidelity measures. *Signal Processing Magazine, IEEE*, 26(1):98–117, 2009.

[2] G. Bjontegaard. Improvements of the bd-psnr model (vceg-m35). Technical report, VCEG Meeting (Document of ITU-T Q.6/SG16), Berlin, Germany, July 2008.

[3] O. Lopez, M. Martinez, P. Piñol, M.P. Malumbres, and J. Oliver. E-ltw: An enhanced ltw encoder with sign coding and precise rate control. In *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pages 2821–2824, Nov 2009.

[4] Zhenghua Yu, Hong Ren Wu, S. Winkler, and Tao Chen. Vision-model-based impairment metric to evaluate blocking artifacts in digital video. *Proceedings of the IEEE*, 90(1):154–169, 2002.

[5] ISO/IEC 10918-1/ITU-T Recommendation T.81. Digital compression and coding of continuous-tone still image, 1992.

[6] W. Pennebaker and J. Mitchell. *JPEG Still Image Data Compression Standard*. Van Nostrand Reinhold, 1994.

[7] D.A. Huffman. A method for the construction of minimum redundancy codes. In *IRE 40*, pages 1098–1101, 1952.

[8] M. Schindler. Huffman coding. http:// www.compressconsult.com/, October 1998.

[9] T.M. Cover and J.A. Thomas. *Elements of information theory*. Wiley Series in Communications, 1991.

[10] G.G. Langdon. An introduction to arithmetic coding. Technical report, IBM J. Res. Develop. 28:135-149, 1984.

[11] J. Rissanen and G.G. Langdon. Arithmetic coding. Technical report, IBM J. Res. Develop. 23:149-162, 1979.

[12] I.H. Witten, R.M. Neal, and J.G. Cleary. Arithmetic coding for data compression. *Commun. ACM*, 30(6):520–540, 1987.

[13] ITU-T Recommendation T.4. Standardization of group 3 fascimile apparatus for document transmission, 1993.

[14] C.R. Hauf and J.C. Houchin. The FlashPix(TM) image file format. In *Fourth Color Imaging Conference: Color Science, Systems and Applications*, pages 234–238, November 1996.

[15] Adobe. Tag based image file format, revision 6.0, http://www.adobe.com/support/technotes.html/, June 1992.

[16] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies. Image coding using wavelet transform. *IEEE Transaction on Image Processing*, 1(2):205–220, 1992.

[17] R.R. Coifman and M.V. Wickerhauser. Entropy-based algorithms for best basis selection. *IEEE Transactions on Information Theory*, 38:713–718, 1992.

[18] K. Ramchandran and M. Vetterli. Best wavelet packet bases in a rate-distortion sense. *IEEE Transactions on Image Processing*, 2:160–175, 1993.

[19] Z. Xiong, K. Ramchandran, and M.T. Orchart. Wavelet packet image coding using space-frequency quantization. *IEEE Transactions on Image Processing*, 7:892–898, June 1998.

[20] F.G. Meyer, A.Z. Averbuch, and J.O. Strömberg. Fast adaptive wavelet packet image compression. *IEEE Transactions on Image Processing*, 9:792–800, May 2000.

[21] N. Sprljan, S. Grgic, M. Mrak, and M. Grgic. Modified SPIHT algorithm for wavelet packet image coding. In *International Symposium on Video/Image Processing and Multimedia Communications (VIProm-Com)*, 2002.

[22] N.M. Rajpoot, R.G. Wilson, F.G. Meyer, and R.R. Coifman. Adaptive wavelet packet basis selection for zerotree image coding. *IEEE Transactions on Image Processing*, 12:1460–1472, December 2003.

[23] Y. Shoham and A. Gersho. Efficient bit allocation for an arbitrary set of quantizers. *IEEE Transactions on Acoustic, Speech, Signal Processing*, 36:1445–1453, September 1984.

[24] J.M. Shapiro. Embedded image coding using zerotrees of wavelet coefficients. *IEEE Transactions on Signal Processing*, 41(12), December 1993.

[25] A. Said and A. Pearlman. A new, fast, and efficient image codec based on set partitioning in hierarchical trees. *IEEE Transactions on circuits and systems for video technology*, 6(3), June 1996.

[26] E.A.B. Da Silva, D.G. Sampson, and M. Ghanbari. A successive approximation vector quantizer for wavelet transform image coding. *IEEE Transactions on Image Processing*, 5:299–310, February 1996.

[27] D. Mukherjee and S.K. Mitra. Successive refinement lattice vector quantization. *IEEE Transactions on Image Processing*, 11:1337–1348, December 2002.

[28] D. Taubman. High performance scalable image compression with EBCOT. *IEEE Transactions on Image Processing*, 9(7):1158–1170, July 2000.

[29] W.A. Pearlman, A. Islam, N. Nagaraj, and A. Said. low-complexity image coding with a set-partitioning embedded block coder. *IEEE Transactions on Circuits and Systems for Video technology*, 14:1219–1235, November 2004.

[30] Stephane Mallat and Sifen Zhong. Characterization of signals from multiscale edges. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(7):710–732, July 1992.

[31] Z. Xiong, K. Ramchandran, and M. Orchard. Space-frequency quantization for wavelet image coding. *IEEE Transactions on Image Processing*, 6(5):677–693, May 1997.

[32] R.L. Joshi, V.J. Crump, and T.R. Fischer. Image subband coding using arithmetic coded trellis coded quantization. *IEEE Transactions on Circuits and Systems for Video technology*, 5, December 1995.

[33] J.W. Woods and S. O'Neil. Subband coding of images. *IEEE Transactions on Acoustic, Speech, Signal Processing*, 34:1278–1288, October 1986.

[34] M.L. Hilton, B.D. Jawerth, and A. Sengupta. Compressing still and moving images with wavelets. *Multimedia Systems*, 2, 1994.

[35] C. Chrysafis and A. Ortega. Line-based, reduced memory, wavelet image compression. *IEEE Transactions on Image Processing*, 9(3):378–389, March 2000.

[36] D. Marpe and H. Cycon. Very low bit-rate video coding using wavelet-based techniques. *IEEE Transactions on Circuits and Systems for Video Technology*, 9:85–94, February 1999.

[37] B.J. Kim, Z. Xiong, and W.A. Pearlman. Low bit-rate scalable video coding with 3D set partitioning in hierarchical trees (3D SPIHT). *IEEE Transactions on Circuits and Systems for Video Technology*, 10:1374–1387, December 2000.

[38] X. Tang, S. Cho, and W.A. Pearlman. Comparison of 3D set partitioning methods in hyperspectral image compression featuring an improved 3D-SPIHT. In *Data Compression Conference*, March 2003.

[39] E.S. Hong and R.E. Ladner. Group testing for image compression. *IEEE Transactions on Image Processing*, 11:901–911, August 2002.

[40] J. Oliver and M. P. Malumbres. Low-complexity multiresolution image compression using wavelet lower trees. *IEEE Transactions on Circuits and Systems for Video Technology*, 16(11):1437–1444, 2006.

[41] H. Everett. Generalized Lagrange multiplier method for solving problems of optimum allocation of resources. *Operations Research*, 11:399–417, 1963.

[42] W.A. Pearlman. Trends of tree-based, set partitioning compression techniques in still and moving image systems. In *Picture Coding Symposium*, pages 1–8, April 2001.

[43] Yushin Cho, W.A. Pearlman, and A. Said. Low complexity resolution progressive image coding algorithm: PROGRESS (progressive resolution decompression). In *IEEE International Conference on Image Processing*, September 2005.

[44] D. Taubman and A. Zakhor. Multirate 3-D subband coding of video. *IEEE Transactions on Image Processing*, 3(5):572–588, September 1994.

[45] ISO/IEC 15444-1. JPEG2000 image coding system, 2000.

[46] D.S. Taubman and M.W. Marcellin. *JPEG 2000: Image Compression Fundamentals, Standards and Practice*, pages 262–281. Kluwer Academic Publishers, 2002.

[47] T. Acharya and P. Tsai. *JPEG 2000 Standard for Image Compression: Concepts, Algorithms and VLSI Arquitectures*, chapter 5. Wiley, October 2005.

[48] M. Rabbani and R. Joshi. An overview of the JPEG2000 still image compression standard. *Signal Processing: Image Communication*, 17, 2002.

[49] M. Albanesi and S. Bertoluzza. Human vision model and wavelets for high-quality image compression. In *International Conference in Image Processing and its Applications*, July 1995.

[50] W. Zeng, S. Daly, and S. Lei. An overview of the visual optimization tools in JPEG2000. *Signal Processing: Image Communication*, 17, 2002.

[51] M.W. Marcellin, M.A. Lepley, A. Bilgin, T.J. Flohr, T.T. Chinen, and J.H. Kasner. An overview of quantization in JPEG2000. *Signal Processing: Image Communication*, 17, 2002.

[52] M.J. Slattery and J.L. Mitchell. The qx-coder. *IBM Journal of Research and Development*, 42:767–784, November 1998.

[53] ITU-T Recommendation H.261. Video codec for audiovisual services at p x 64 kbits/s, March 1999.

[54] M. Liu. Overview of the p * 64 kbits/s video coding standard. *Communications ACM*, 34(4):60–63, April 1991.

[55] ISO/IEC JTC1. ISO/IEC 11172-2. Coding of moving pictures and associated audio for digital storage media at up to about 1.5 mbit/s, 1993.

[56] ISO/IEC JTC1. ISO/IEC 13818-2. Generic coding of moving pictures, 2000.

[57] ISO/IEC JTC1. ISO/IEC 14496-2. Coding of audio-visual objects, April 2001.

[58] T. Sikora. MPEG digital video-coding standard. *IEEE Signal Processing Magazine*, September 1997.

[59] A. Hallapuro, M. Karczewicz, and H. Malvar. Low complexity transform and quantization - part i: Basic implementation. Technical report, Tech. Report JVTB038, Joint Video Team (JVT), February 2002.

[60] ISO/IEC 14496-10 and ITU Rec. H.264. Advanced video coding, 2003.

[61] I.E.G. Richardson. *H.264 and MPEG-4 Video Compression*. John Wiley and Sons Ltd, 2003.

[62] G. Björntegaard and K. Lillevold. Content-adaptative VLC coding and coefficients. Technical Report JVT-C028, Joint Video Team (JVT), May 2002.

[63] Information technology. coded representation of picture and audio information - progressive bi-level image compression. Technical report, Tech. Report T.82 (JBIG), ITU-T Recommendation.

[64] D. Marpe, G. Blättermann, and T. Wiegand. Adaptive codes for H.26L. Technical Report VCEG-L13, ITU-T SG16/6, Eibsee, Germany, January 2001.

[65] M. Karczewicz and R. Kurceren. A proposal for SP-frames. Technical report, VCEG-L27, ITU-T SG 16/6, January 2001.

[66] M. Karczewicz and R. Kurceren. The SP and SI frames design for H.264/AVC. *IEEE Transactions on Circuits and System for Video Technology*, 13:637–644, 2003.

[67] G.J. Sullivan, J. Ohm, Woo-Jin Han, and T. Wiegand. Overview of the high efficiency video coding (hevc) standard. *Circuits and Systems for Video Technology, IEEE Transactions on*, 22(12):1649–1668, Dec 2012.

[68] M.T. Pourazad, C. Doutre, M. Azimi, and P. Nasiopoulos. Hevc: The new gold standard for video compression: How does hevc compare with h.264/avc? *Consumer Electronics Magazine, IEEE*, 1(3):36–46, July 2012.

[69] G. Karlsson and M. Vetterli. Three-dimensional subband coding of video. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 2, pages 1100–1103, April 1988.

[70] B.J. Kim and W.A. Pearlman. An embedded wavelet video coder using three-dimensionnal set partitionning in hierarchical trees (SPIHT). In *Data Compression Conference*, pages 251–260, March 1997.

[71] J. Tham, S. Ranganath, and A. Kassim. Highly scalable wavelet-based video codec for very low bit-rate environment. *IEEE Journal on Selected Areas in Communications*, 16:12–27, January 1998.

[72] J.R. Ohm. Advanced packet-video coding based on layered VQ and SBC techniques. *IEEE Transactions on Circuits and Systems for Video Technology*, 3(3):208–221, June 1994.

[73] J.R. Ohm. Three dimensional subband coding with motion compensation. *IEEE Transactions on Image Processing*, 3(5):559–571, September 1994.

[74] S.J. Choi and J.W.Woods. Motion-compensated 3-d subband coding of video. *IEEE Transactions on Image Processing*, 8(2):155–167, February 1999.

[75] J.W. Woods and G. Lilienfield. A resolution and frame-rate scalable subband/wavelet video coder. *IEEE Transactions on Circuits and Systems for Video Technology*, pages 1035–1044, September 2001.

[76] A. Secker and D. Taubman. Motion-compensated highly scalable video compression using an adaptive 3D wavelet transform based on lifting. *IEEE Internantional Conference on Image Processing*, pages 1029–1032, October 2001.

[77] A. Secker and D. Taubman. Highly scalable video compression using a lifting-based 3D wavelet transform with deformable mesh motion compensation. *IEEE Internantional Conference on Image Processing*, pages 749–752, September 2002.

[78] A. Secker and D. Taubman. Highly scalable video compression with scalable motion coding. *IEEE Internantional Conference on Image Processing*, September 2003.

[79] A. Secker and D. Taubman. Lifting-based invertible motion adaptive transform (LIMAT) framework for highly scalable video compression. *IEEE Transactions on Image Processing*, 12(12):1530–1542, December 2003.

[80] B. Pesquet-Popescu and V. Bottreau. Three-dimensional lifting schemes for motion compensated video compression. *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1793–1796, 2001.

[81] Zhou Wang, A.C. Bovik, and Ligang Lu. Why is image quality assessment so difficult? In *Acoustics, Speech, and Signal Processing (ICASSP), 2002 IEEE International Conference on*, volume 4, pages IV–3313–IV–3316, 2002.

[82] ITU Telecomunication Standardization Sector of ITU. Itu-r bt.500-11 methodology for the subjective assessment of the quality of television pictures. Technical report, ITU, 2002.

[83] ITU Telecomunication Standardization Sector of ITU. Itu-r bt.500-12 methodology for the subjective assessment of the quality of television pictures - ihs, inc. Technical report, ITU, 2009.

[84] ITU Telecomunication Standardization Sector of ITU. Tu-t p.910 (04/2008) subjective video quality assessment methods for multimedia applications, 2008.

[85] Hamid Rahim Sheikh, Alan Conrad Bovik, and Gustavo de Veciana. An information fidelity criterion for image quality assessment using natural scene statistics. *IEEE Transactions on Image Processing*, 14(12), 2005.

[86] A.M. van Dijk and J.-B. Martens. Quality assessment of compressed images: A comparison between two methods. In *Image Processing, 1996. Proceedings., International Conference on*, volume 1, pages 25–28 vol.2, 1996.

[87] K.T. Tan, M. Ghanbari, and D.E. Pearson. An objective measurement tool for {MPEG} video quality. *Signal Processing*, 70(3):279 – 294, 1998.

[88] Margaret H. Pinson and Stephen Wolf. Comparing subjective video quality testing methodologies. *Proc. SPIE 5150, Visual Communications and Image Processing 2003*, pages 573–582, June 2003.

[89] H. R. Wu and K. R. Rao. *Digital Video Image Quality and Perceptual Coding (Signal Processing and Communications)*. CRC Press, Inc., Boca Raton, FL, USA, 2005.

[90] Philip Corriveau, Christina Gojmerac, Bronwen Hughes, and Lew Stelmach. All subjective scales are not created equal: The effects of context on different scales. *Signal Processing*, 77(1):1 – 9, 1999.

[91] Thrasyvoulos N. Pappas and Robert J. Safranek. Perceptual criteria for image quality evaluation. In *in Handbook of Image and Video Processing*, pages 669–684. Academic Press, 2000.

[92] Zhou Wang and Alan C. Bovik. *Modern Image Quality Assessment*. Synthesis Lectures on Image, Video, and Multimedia Processing. Morgan & Claypool Publishers, 2006.

[93] K. Seshadrinathan, R. Soundararajan, A.C. Bovik, and L.K. Cormack. Study of subjective and objective quality assessment of video. *Image Processing, IEEE Transactions on*, 19(6):1427–1441, 2010.

[94] J. Korhonen and Junyong You. Peak signal-to-noise ratio revisited: Is simple beautiful? In *Quality of Multimedia Experience (QoMEX), 2012 Fourth International Workshop on*, pages 37–38, 2012.

[95] S. Winkler. Issues in vision modeling for perceptual video quality assessment. *Signal Processing*, 78(2), 1999.

[96] Z. Wang, H. R. Sheikh, and A. C. Bovik. *The Handbook of Video Databases: Design and Applications*, chapter 41 Objective Video Quality Assessment, pages 1041–1078. CRC Press, 2003.

[97] K. Brunnstrom, D. Hands, F. Speranza, and A. Webster. Vqeg validation and itu standardization of objective perceptual video quality metrics [standards in a nutshell]. *Signal Processing Magazine, IEEE*, 26(3):96–101, 2009.

[98] Farzad Ebrahimi, Matthieu Chamik, and Stefan Winkler. Jpeg vs. jpeg2000: An objective comparison of image encoding quality. In *Proceedings of SPIE Applications of Digital Image Processing*, page 300308, 2004.

[99] Michael Yuen and H.R. Wu. A survey of hybrid mc/dpcm/dct video coding distortions. *Signal Processing*, 70(3):247 – 278, 1998.

[100] R. Leung and D. Taubman. Minimizing the perceptual impact of visual distortion in scalable wavelet compressed video. In *Image Processing, 2006 IEEE International Conference on*, pages 633–636, 2006.

[101] Ying Luo and Rabab K. Ward. Removing the blocking artifacts of block-based dct compressed images. *IEEE Transactions on Image Processing*, 12(7), July 2003.

[102] R. Kakarala and R. Bagadi. A method for signalling block-adaptive quantization in baseline sequential jpeg. In *TENCON 2009 - 2009 IEEE Region 10 Conference*, pages 1–6, 2009.

[103] KinTak U., Nian Ji, Dongxu Qi, and Zesheng Tang. An adaptive quantization technique for jpeg based on non-uniform rectangular partition. In

Ying Zhang, editor, *Future Wireless Networks and Information Systems*, volume 143 of *Lecture Notes in Electrical Engineering*, pages 179–187. Springer Berlin Heidelberg, 2012.

[104] Quoc Bao Do, M. Luong, and A. Beghdadi. A new perceptually adaptive method for deblocking and deringing. In *Information Science, Signal Processing and their Applications (ISSPA), 2012 11th International Conference on*, pages 533–538, 2012.

[105] W. Li, O. Egger, and M. Kunt. Efficient quantization noise reduction device for subband image coding schemes. In *Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on*, volume 4, pages 2209–2212 vol.4, 1995.

[106] Andrew B. Watson, Gloria Y. Yang, Joshua A. Solomon, and John Villasenor. Visibility of wavelet quantization noise. *IEEE TRANSACTIONS ON IMAGE PROCESSING*, 6(8):1164–1175, 1997.

[107] Ngai-Fong Law, Wan-Chi Siu, and Degan Feng. Suppression of ringing artifacts with an adaptive shrinkage algorithm. In *Communications, Computers and Signal Processing, 1999 IEEE Pacific Rim Conference on*, pages 181–184, 1999.

[108] V.K. Nath and D. Hazarika. Blocking artifacts suppression in wavelet transform domain using local wiener filtering. In *Emerging Trends and Applications in Computer Science (NCETACS), 2012 3rd National Conference on*, pages 93–97, 2012.

[109] J. Oliver and M.P. Malumbres. Fast and efficient spatial scalable image compression using wavelet lowertrees. In *IEEE Data Compression Conference*, Snowbird, UT, 2003.

[110] Marcus J Nadenau, Stefan Winkler, David Alleysson, and Murat Kunt. Human vision models for perceptually optimized image processing–a review. *Proceedings of the IEEE*, page 32, 2000.

[111] Marcus Nadenau. *Integration of human color vision models into high quality image compression*. PhD thesis, STI, Lausanne, 2000.

[112] L. K. Cormack. *Handbook of Image and Video Processing*, chapter 4.1 Computational models, of early human vision, pages 271–287. Academic Press, 2000.

[113] G. Westheimer. *Handbook of Perception and Human Performance*, volume Vol.1 Chap. 4, chapter The eye as an optical instrument. John Wiley & Sons, 1986.

[114] David R. Williams. Topography of the foveal cone mosaic in the living human eye. *Vision Research*, 28(3):433 – 454, 1988.

[115] Jeffrey Lubin. Digital images and human vision. chapter The use of psychophysical data and models in the analysis of display system performance, pages 163–178. MIT Press, Cambridge, MA, USA, 1993.

[116] Sanghoon Lee, M.S. Pattichis, and A.C. Bovik. Foveated video quality assessment. *Multimedia, IEEE Transactions on*, 4(1):129–132, 2002.

[117] Zhou Wang and A.C. Bovik. Embedded foveation image coding. *Image Processing, IEEE Transactions on*, 10(10):1397–1410, 2001.

[118] Michael P. Eckert and Andrew P. Bradley. Perceptual quality metrics applied to still image compression. *Signal Processing*, 70(3):177 – 200, 1998.

[119] D.C. Hood and M.A. Finkelstein. *Sensitivity to light*, volume 1, chapter Handbook of Perception and Human Performance. 1986.

[120] Randolph Blake and Rober Sekuler. *Perception*, chapter Ch5. Spatial Vision and Form Perception, pages 151–192. 2005.

[121] F.W. Campbell and J.G. Robson. Application of fourier analysis to the visibility of gratings. *Jounal of Physiology*, 197:551–566, 1968.

[122] F.W. Campbell and C Blakemore. On the existence of neurones in the human visual system selectively sensitive to the orientation and size of retinal images. *Journal of Physiology (London)*, 203:237–260, 1969.

[123] Selig Hecht. The visual discrimination of intensity and the weber-fechner law. *Journal of General Psychology*, 7(2):235–267, 1924.

[124] Kathy T. Mullen. The contrast sensitivity of human colour vision to red-green and blue-yelow chromatic gratings. *Journal of Physiology*, pages 381–400, 1985.

[125] Stefan Winkler. *Digital video quality: vision models and metrics*. Wiley, 2005.

[126] S. Daly. Engineering observations from spatiovelocity and spatiotemporal visual models. In *Proc. SIPIE*, volume 3299, pages 180–191, San Jose, CA, 1998.

[127] Jan J. Koenderink and Andrea J. van Doorn. Spatiotemporal contrast detection threshold surface is bimodal. *Opt. Lett.*, 4(1):32–34, Jan 1979.

[128] J. G. ROBSON. Spatial and temporal contrast-sensitivity functions of the visual system. *J. Opt. Soc. Am.*, 56(8):1141–1142, Aug 1966.

[129] D. H. Kelly. Spatiotemporal variation of chromatic and achromatic contrast thresholds. *J. Opt. Soc. Am.*, 73(6):742–749, Jun 1983.

[130] Jian Yang and Walter Makous. Spatiotemporal separability in contrast sensitivity. *Vision Research*, 34(19):2569 – 2576, 1994.

[131] Dawei W. Dong. Spatiotemporal inseparability of natural images and visual sensitivities. In *In Computational, Neural & Ecological Constraints of Visual Motion Processing, J.M. Zanker & J. Zeil (Eds*, pages 371–380. Springer Verlag, 1999.

[132] M. A. Georgeson and G. D. Sullivan. Contrast constancy: Deblurring in human vision by spatial frequency channels. *Journal of Physiology*, 252(3):627–656, 1975.

[133] J. Fiser, P. J. Bex, and W. Makous. Contrast conservation in human vision. *Vision Research*, 43(25):2637–48, 2003.

[134] Michael A. Webster and Eriko Miyahara. Contrast adaptation and the spatial structure of natural images. *J. Opt. Soc. Am. A*, 14(9):2355–2366, Sep 1997.

[135] Damon M. Chandler and Sheila S. Hemami. Suprathreshold image compression based on contrast allocation and global precedence. In *in Proc. SPIE Human Vision and Electronic Imaging VIII*, pages 73–86, 2003.

[136] VQEG. Final report from the video quality experts group on the validation of objective models of video quality assessment. phase I, Marz 2000.

[137] VQEG. Final report from the video quality experts group on the validation of objective models of video quality assessment. phase II, August 2003.

[138] S. Winkler and P. Mohandas. The evolution of video quality measurement: From psnr to hybrid metrics. *Broadcasting, IEEE Transactions on*, 54(3):660–668, 2008.

[139] F. Porikli, A. Bovik, C. Plack, G. AlRegib, J. Farrell, P. Le Callet, Quan Huynh-Thu, S. Moller, and S. Winkler. Multimedia quality assessment [dsp forum]. *Signal Processing Magazine, IEEE*, 28(6):164–177, 2011.

[140] Stephen Wolf and Margaret H. Pinson. Spatial-temporal distortion metric for in-service quality monitoring of any digital video system, 1999.

[141] M. Masry, S. S. Hemami, and Y. Sermadevi. A scalable wavelet-based video distortion metric and applications. *IEEE Trans. Cir. and Sys. for Video Technol.*, 16(2):260–273, September 2006.

[142] ITU Telecomunication Standardization Sector of ITU. Itu-t rec j.144. objective perceptual video quality measurement techniques for digital cable television in the presence of a full reference, March 2001.

[143] Video Quality Experts Group (VQEG). Final report from the video quality experts group on the validation of objective models of multimedia quality assessment, phase i, March 2008.

[144] ITU Telecomunication Standardization Sector of ITU. Itu-t rec j.247. objective perceptual multimedia video quality measurement in the presence of a full reference, August 2008.

[145] ITU Telecomunication Standardization Sector of ITU. Itu-t rec j.246. perceptual audiovisual quality measurement techniques for multimedia services over digital cable television networks in the presence of a reduced bandwidth reference, August 2008.

[146] U. Engelke and H-J Zepernick. Perceptual-based quality metrics for image and video services: A survey. In *Next Generation Internet Networks, 3rd EuroNGI Conference on*, pages 190–197, 2007.

[147] S. Winkler. Video quality measurement standards current status and trends. In *Information, Communications and Signal Processing, 2009. ICICS 2009. 7th International Conference on*, pages 1–5, 2009.

[148] S. Chikkerur, V. Sundaram, M. Reisslein, and L.J. Karam. Objective video quality assessment methods: A classification, review, and performance comparison. *Broadcasting, IEEE Transactions on*, 57(2):165–182, 2011.

[149] Weisi Lin and C.-C. Jay Kuo. Perceptual visual quality metrics: A survey. *Journal of Visual Communication and Image Representation*, 22(4):297 – 312, 2011.

[150] P.C. Teo and D.J. Heeger. Perceptual image distortion. In *Image Processing, 1994. Proceedings. ICIP-94., IEEE International Conference*, volume 2, pages 982–986 vol.2, 1994.

[151] Christian J. van den Branden Lambrecht and Olivier Verscheure. Perceptual quality measure using a spatiotemporal model of the human visual system. In *Storage and Retrieval for Image and Video Databases*, volume 2668, pages 450–461, 1996.

[152] Andrew B. Watson, James Hu, and John F Mcgowan Iii. Dvq: A digital video quality metric based on human vision. *Journal of Electronic Imaging*, 10:20–29, 2001.

[153] J. Malo, A.M. Pons, and J.M. Artigas. Subjective image fidelity metric based on bit allocation of the human visual system in the {DCT} domain. *Image and Vision Computing*, 15(7):535 – 548, 1997.

[154] Andrew B Watson. Toward a perceptual video-quality metric. In *Photonics West'98 Electronic Imaging*, pages 139–147. International Society for Optics and Photonics, 1998.

[155] Zhou Wang, Alan C. Bovik, Ligang Lu, and Jack L. Kouloheris. Foveated wavelet image quality index, 2001.

[156] A. Cavallaro and S. Winkler. Segmentation-driven perceptual quality metrics. In *Image Processing, 2004. ICIP '04. 2004 International Conference on*, volume 5, pages 3543–3546 Vol. 5, 2004.

[157] Stefan Winkler. Perceptual distortion metric for digital color video, 1999.

[158] Stefan Winkler. Quality metric design: a closer look, 2000.

[159] Eli Peli. Contrast in complex images. *J. Opt. Soc. Am. A*, 7(10):2032–2040, Oct 1990.

[160] C.J. Van Den Branden Lambrecht. A working spatio-temporal model of the human visual system for image restoration and quality assessment applications. In *Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on*, volume 4, pages 2291–2294 vol. 4, 1996.

[161] Andrew B. Watson. The cortex transform: Rapid computation of simulated neural images. *Computer Vision, Graphics, and Image Processing*, 39(3):311 – 327, 1987.

[162] Scott Daly. Digital images and human vision. chapter The visible differences predictor: an algorithm for the assessment of image fidelity, pages 179–206. MIT Press, Cambridge, MA, USA, 1993.

[163] E. P. Simoncelli, W. T. Freeman, E. H. Adelson, and D. J. Heeger. Shiftable multiscale transforms. *IEEE Trans. Inf. Theor.*, 38(2):587–607, September 2006.

[164] Andrew B Watson. Visual optimization of dct quantization matrices for individual images. In *Proc. AIAA Computing in Aerospace*, volume 9, pages 286–291, 1993.

[165] Marcus J. Nadenau, Julien Reichel, and Murat Kunt. Performance comparison of masking models based on a new psychovisual test method with natural scenery stimuli. *Signal Processing: Image Communication*, 17(10):807 – 823, 2002.

[166] Andrew B Watson and Joshua A Solomon. Model of visual contrast gain control and pattern masking. *JOSA A*, 14(9):2379–2391, 1997.

[167] A.B. Watson. Perceptual optimization of dct color quantization matrices. In *Image Processing, 1994. Proceedings. ICIP-94., IEEE International Conference*, volume 1, pages 100–104 vol.1, 1994.

[168] Stefan Winkler and Ruth Campos. Video quality evaluation for internet streaming applications, 2003.

[169] Y. Sermadevi and S.S. Hemami. Linear programming optimization for video coding under multiple constraints. In *Data Compression Conference, 2003. Proceedings. DCC 2003*, pages 53–62, 2003.

[170] Arthur A Webster, Coleen T Jones, Margaret H Pinson, Stephen D Voran, and Stephen Wolf. Objective video quality assessment system based on human perception. In *IS&T/SPIE's Symposium on Electronic Imaging: Science and Technology*, pages 15–26. International Society for Optics and Photonics, 1993.

[171] Stephen Wolf and Margaret Pinson. Technical report tr-02-392 - video quality measurement techniques. Technical report, National Telecommunications & Information Administration, 2002.

[172] M.H. Pinson and S. Wolf. A new standardized method for objectively measuring video quality. *Broadcasting, IEEE Transactions on*, 50(3):312–322, 2004.

[173] Stephen Wolf and Margaret H Pinson. Low bandwidth reduced reference video quality monitoring system. In *First Int'l Workshop on Video Proc. and Quality Metrics*, 2005.

[174] Zhou Wang, A.C. Bovik, and B.L. Evan. Blind measurement of blocking artifacts in images. In *Image Processing, 2000. Proceedings. 2000 International Conference on*, volume 3, pages 981–984 vol.3, 2000.

[175] H.R. Wu and M. Yuen. A generalized block-edge impairment metric for video coding. *Signal Processing Letters, IEEE*, 4(11):317–320, 1997.

[176] A.C. Bovik and Shizhong Liu. Dct-domain blind measurement of blocking artifacts in dct-coded images. In *Acoustics, Speech, and Signal Processing, 2001. Proceedings. (ICASSP '01). 2001 IEEE International Conference on*, volume 3, pages 1725–1728 vol.3, 2001.

[177] Shizhong Liu and A.C. Bovik. Efficient dct-domain blind measurement and reduction of blocking artifacts. *Circuits and Systems for Video Technology, IEEE Transactions on*, 12(12):1139–1149, 2002.

[178] S.A. Karunasekera and N.G. Kingsbury. A distortion measure for blocking artifacts in images based on human visual sensitivity. *Image Processing, IEEE Transactions on*, 4(6):713–724, 1995.

[179] Zhou Wang, Hamid R. Sheikh, and A.C. Bovik. No-reference perceptual quality assessment of jpeg compressed images. In *Image Processing. 2002. Proceedings. 2002 International Conference on*, volume 1, pages I–477–I–480 vol.1, 2002.

[180] P. Marziliano, F. Dufaux, S. Winkler, and T. Ebrahimi. A no-reference perceptual blur metric. In *Image Processing. 2002. Proceedings. 2002 International Conference on*, volume 3, pages III–57–III–60 vol.3, 2002.

[181] Pina Marziliano, Frederic Dufaux, Stefan Winkler, and Touradj Ebrahimi. Perceptual blur and ringing metrics: application to jpeg2000. *Signal Processing: Image Communication*, 19(2):163–172, 2004.

[182] T.M. Kusuma and H-J Zepernick. A reduced-reference perceptual quality metric for in-service image quality assessment. In *Mobile Future and Symposium on Trends in Communications, 2003. SympoTIC '03. Joint First Workshop on*, pages 71–74, 2003.

[183] S. Saha and R. Vemuri. Effect of image activity on lossy and lossless coding performance. In *Data Compression Conference, 2000. Proceedings. DCC 2000*, pages 570–, 2000.

[184] Paolo Gastaldo, Rodolfo Zunino, Ingrid Heynderickx, and Elena Vicario. Objective quality assessment of displayed images by using neural networks. *Signal Processing: Image Communication*, 20(7):643–661, 2005.

[185] Marco Montenovo, Alessandro Perot, Marco Carli, Paolo Cicchetti, and Alessandro Neri. Objective quality evaluation of video services. In *Sec-*

ond International Workshop on Video Processing and Quality Metrics for Consumer Electronics, 2006.

[186] S. Winkler, E. D. Gelasca, and T. Ebrahimi. Perceptual quality assessment for video watermarking. In *International Conference on Information Technology: Coding and Computing, 2002. Proceedings.*, pages 90–94, April 2002.

[187] Zhou Wang, Guixing Wu, Hamid R. Sheikh Member, Eero P. Simoncelli Senior Member, En hui Yang, Senior Member, and Alan C. Bovik. Quality-aware images. *IEEE Transactions on Image Processing*, 15:1680–1689, 2006.

[188] S. J. P. Westen, R. L. Lagendijk, and J. Biemond. Optimization of jpeg color image coding using a human visual system model. In *SPIE conference on Human Vision and Electronic Imaging*, pages 370–381, 1996.

[189] Peter G. J. Barten. Evaluation of subjective image quality with the square-root integral method. *J. Opt. Soc. Am. A*, 7(10):2024–2031, Oct 1990.

[190] Russel A. Martin, Albert J. Ahumada, Jr., and James O. Larimer. Color matrix display simulation based on luminance and chromatic contrast sensitivity of early vision, 1992.

[191] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *Image Processing, IEEE Transactions on*, 13(4):600–612, 2004.

[192] Andrew B. Watson and Lindsay Kreslake. Measurement of visual impairment scales for digital video, 2001.

[193] Marcia G. Ramos and Sheila S. Hemami. Suprathreshold wavelet coefficient quantization in complex stimuli: psychophysical evaluation and analysis. *J. Opt. Soc. Am. A*, 18(10):2385–2397, Oct 2001.

[194] Damon M. Chandler and Sheila S. Hemami. Additivity models for suprathreshold distortion in quantized wavelet-coded images, 2002.

[195] A. Srivastava, A. B. Lee, E. P. Simoncelli, and S.-C. Zhu. On advances in statistical modeling of natural images. *J. Math. Imaging Vis.*, 18(1):17–33, January 2003.

[196] Zhou Wang and A.C. Bovik. A universal image quality index. *Signal Processing Letters, IEEE*, 9(3):81–84, 2002.

[197] Zhou Wang, Alan C. Bovik, and Eero P. Simoncelli. *Handbook of Image and Video Processing*, chapter 8.3 Structural Approaches to Image Quality Assessment. Aca, 2005.

[198] Z. Wang, L. Lu, and A. Bovik. Video quality assessment using structural distortion measurement. In *Proceedings IEEE International Conference of Image Processing*, volume 3, pages 65–68, September 2002.

[199] Zhou Wang, Ligang Lu, and Alan C. Bovik. Video quality assessment based on structural distortion measurement. *Signal Processing: Image Communication, special issue on "Objective Video Quality Metrics"*, 19(2):121–132, February 2004.

[200] Z. Wang, E.P. Simoncelli, and A.C. Bovik. Multiscale structural similarity for image quality assessment. In *Signals, Systems and Computers, 2004. Conference Record of the Thirty-Seventh Asilomar Conference on*, volume 2, pages 1398–1402 Vol.2, 2003.

[201] Zhou Wang and E.P. Simoncelli. Translation insensitive image similarity in complex wavelet domain. In *Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP '05). IEEE International Conference on*, volume 2, pages 573–576, 2005.

[202] Zhou Wang and Eero P. Simoncelli. An adaptative linear system framework for image distortion analysis. In *Proc. 12th IEEE Intl. Conf. Image Processing Vol III, pp 1160-1163, Sep 2005.*, 2005.

[203] Eero P Simoncelli. Modeling the joint statistics of images in the wavelet domain. In *SPIE's International Symposium on Optical Science, Engineering, and Instrumentation*, pages 188–195. International Society for Optics and Photonics, 1999.

[204] Zhou Wang and Eero P. Simoncelli. Reduced-reference image quality assessment using a wavelet-domain natural image statistic model. In *Human Vision and Electronic Imaging X, Proc. SPIE, vol. 5666.*, 2005.

[205] H.R. Sheikh, A.C. Bovik, and L. Cormack. No-reference quality assessment using natural scene statistics: Jpeg2000. *Image Processing, IEEE Transactions on*, 14(11):1918–1927, 2005.

[206] E.P. Simoncelli. Statistical models for images: compression, restoration and synthesis. In *Signals, Systems amp; Computers, 1997. Conference Record of the Thirty-First Asilomar Conference on*, volume 1, pages 673–678 vol.1, 1997.

[207] R.W. Buccigrossi and E.P. Simoncelli. Image compression via joint statistical characterization in the wavelet domain. *Image Processing, IEEE Transactions on*, 8(12):1688–1701, 1999.

[208] Martin J Wainwright, Eero P Simoncelli, and Alan S Willsky. Random cascades on wavelet trees and their use in analyzing and modeling natural images. *Applied and Computational Harmonic Analysis*, 11(1):89–123, 2001.

[209] J. Korhonen, N. Burini, Junyong You, and E. Nadernejad. How to evaluate objective video quality metrics reliably. In *Quality of Multimedia Experience (QoMEX), 2012 Fourth International Workshop on*, pages 57–62, 2012.

[210] H. R. Sheikh, M. F. Sabir, and A. C. Bovik. A statistical evaluation of recent full reference image quality assessment algorithms. *IEEE Transactions on Image Processing*, 15(11):3440– 3451, 2006.

[211] H.R. Sheikh, Z.Wang, L. Cormack, and A.C. Bovik. Live image quality assessment database release 2. http://live.ece.utexas.edu/research/quality.

[212] Eric C. Larson and Damon M. Chandler. Most apparent distortion: full-reference image quality assessment and the role of strategy. *Journal of Electronic Imaging*, 19(1):011006, 2010.

[213] Patrick Le Callet and Florent Autrusseau. Subjective quality assessment irccyn/ivc database, 2005. http://www.irccyn.ec-nantes.fr/ivcdb/.

[214] Media Information and Communications Technology Laboratory. Toyama image database. OnLine - http://160.26.142.130/mictdb.html, 2010.

[215] D.M. Chandler and S.S. Hemami. Vsnr: A wavelet-based visual signal-to-noise ratio for natural images. *Image Processing, IEEE Transactions on*, 16(9):2284–2298, Sept 2007.

[216] Nikolay Ponomarenko, Federica Battisti, Karen Egiazarian, Jaakko Astola, and Vladimir Lukin. Metrics performance comparison for color image database. In *Fourth international workshop on video processing and quality metrics for consumer electronics*, volume 27, 2009.

[217] U. Engelke, H.J. Zepernick, and M. Kusuma. Wireless imaging quality database. OnLine - http://www.bth.se/tek/rcg.nsf/pages/wiq-db, 2010.

[218] P. V. Vu and D. M. Chandler. Vis3: An algorithm for video quality assessment via analysis of spatial and spatiotemporal slices. *Journal of Electronic Imaging (JEI)*, 23(1), 2014.

[219] Technische Universität München, Institute for Data Processing. TUM LDV Multi Format Test Set, 2011.

[220] Video Quality Experts Group (VQEG). Vqeg fr-tv phase i database. http://www.its.bldrdoc.gov/vqeg/downloads.aspx.

[221] Video Quality Experts Group (VQEG). Vqeg hdtv phase i database. http://www.its.bldrdoc.gov/vqeg/downloads.aspx.

[222] H.R. Sheikh and A.C. Bovik. Image information and visual quality. *Image Processing, IEEE Transactions on*, 15(2):430–444, 2006.

[223] Ann Marie Rohaly, Philip Corriveau, John Libert, Arthur Webster, Vittorio Baroncini, John Beerends, and Jean-Louis Blin. Video quality experts group: Current results and future directions, 2000.

[224] ISO/IEC 14496-10:2003. Coding of audiovisual objects part 10: advanced videocoding. ITUT Recommendation H264 Advanced video codingfor generic audiovisual services, 2003.

[225] ISO/IEC 15444-1. Jpeg 2000 image coding system. part 1:core coding system,, 2000.

[226] J. Oliver and M.P. Malumbres. Low-complexity multiresolution image compression using wavelet lower trees. *IEEE Transactions on Circuits and Systems for Video Technology*, 17(11):1437–1444, Nov 2006.

[227] Carlos T. Calafate, P. Manzoni, and Manuel P. Malumbres. Speeding up the evaluation of multimedia streaming applications in MANETs using HMMs. In *Proceedings of the 7th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems*, pages 315–322, 2004.

[228] IEEE. IEEE 802.11 WG. 802.11e Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements, 2005.

[229] Carlos T. Calafate, Manuel P. Malumbres, and P. Manzoni. Performance of H.264 compressed video streams over 802.11b based MANETs. In *Proceedings of the 24th International Conference on Distributed Computing Systems Workshops - W7: EC (ICDCSW'04) - Volume 7*, pages 776 – 781, 2004.

[230] R.M. Gray and D.L. Neuhoff. Quantization. *Information Theory, IEEE Transactions on*, 44(6):2325–2383, Oct 1998.

[231] J. Max. Quantizing for minimum distortion. *Information Theory, IRE Transactions on*, 6(1):7–12, March 1960.

[232] V. Algazi. Useful approximations to optimum quantization. *Communication Technology, IEEE Transactions on*, 14(3):297–301, June 1966.

[233] Mohammad A. Khan and Mark J. T. Smith. *Handbook of Image and Video Processing*, chapter 5.3 Fundamentals of VectorQuantization, pages 512–521. 2000.

[234] H. R. Wu, K. R. Rao, and Ashraf A. Kassim. Digital video image quality and perceptual coding. *J. Electronic Imaging*, 16(3):039901, 2007.

[235] R.J. Safranek and J.D. Johnston. A perceptually tuned sub-band image coder with image dependent quantization and post-quantization data compression. In *Acoustics, Speech, and Signal Processing, 1989. ICASSP-89., 1989 International Conference on*, pages 1945–1948 vol.3, May 1989.

[236] Tom Cornsweet. *Visual Perception*. Elsevier Science, 2012.

[237] Heidi A. Peterson, Huei Peng, J. H. Morgan, and William B. Pennebaker. Quantization of color image components in the dct domain, 1991.

[238] Antonio Ortega. *Optimization techniques for adaptive quantization of image and video under delay constraints*. PhD thesis, Columbia University, 1994.

[239] Yun Q. Shi and Huifang Sun. *Image and Video Compression for Multimedia Engineering: Fundamentals, Algorithms, and Standards*. CRC Press, Inc., Boca Raton, FL, USA, 2nd edition, 2008.

[240] J. Mannos and D.J. Sakrison. The effects of a visual fidelity criterion of the encoding of images. *Information Theory, IEEE Transactions on*, 20(4):525–536, Jul 1974.

[241] Raghu Machiraju, Ajeetkumar Gaddipatti, and Roni Yagel. Steering image generation with wavelet based perceptual metric. In *Computer Graphics Forum*, pages 241–251.

[242] D.S. Taubman and M.W. Marcellin. *Jpeg2000: Image Compression Fundamentals, Standards, and Practice*. The Springer International Series in Engineering and Computer Science Series. Springer-Verlag GmbH, 2002.

[243] Noureddine Moumkine, Ahmed Tamtaoui, and A Ait Ouahman. Integration of the contrast sensitivity function into wavelet codec. In *In Proc. Second International Symposium on Comunications, Control and Signal Processing ISCCSP, Marrakech, Morocco*, 2006.

[244] A. Bajit, M. Nahid, A. Tamtaoui, and E. H. Bouyakhf. A perceptually optimized wavelet embedded zerotree image coder. *International Journal of Signal Processing*, 4(4):296–301, April 2007.

[245] N. Nill. A visual model weighted cosine transform for image compression and quality assessment. *Communications, IEEE Transactions on*, 33(6):551–557, Jun 1985.

[246] King N.Ngan, K.S. Leong, and H. Singh. Adaptive cosine transform coding of images in perceptual domain. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 37(11):1743–1750, Nov 1989.

[247] B. Chitprasert and K.R. Rao. Human visual weighted progressive image transmission. *Communications, IEEE Transactions on*, 38(7):1040–1044, Jul 1990.

[248] D.M. Chandler and S.S. Hemami. Dynamic contrast-based quantization for lossy wavelet image compression. *Image Processing, IEEE Transactions on*, 14(4):397–410, April 2005.

[249] M.J. Nadenau, J. Reichel, and M. Kunt. Wavelet-based color image compression: exploiting the contrast sensitivity function. *Image Processing, IEEE Transactions on*, 12(1):58–70, Jan 2003.

[250] Albert J. Ahumada, Jr. and Heidi A. Peterson. Luminance-model-based dct quantization for color image compression, 1992.

[251] Heidi A Peterson, Albert J Ahumada Jr, and Andrew B Watson. Improved detection model for dct coefficient quantization. In *IS&T/SPIE's Symposium on Electronic Imaging: Science and Technology*, pages 191–201. International Society for Optics and Photonics, 1993.

[252] Zhen Liu, L.J. Karam, and A.B. Watson. Jpeg2000 encoding with perceptual distortion control. *Image Processing, IEEE Transactions on*, 15(7):1763–1778, July 2006.

[253] H.H.Y. Tong and A.N. Venetsanopoulos. A perceptual model for jpeg applications based on block classification, texture masking, and luminance masking. In *Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on*, pages 428–432 vol.3, Oct 1998.

[254] Gordon E. Legge and John M. Foley. Contrast masking in human vision. *J. Opt. Soc. Am.*, 70(12):1458–1471, Dec 1980.

[255] Wenjun Zeng, Scott Daly, and Shawmin Lei. An overview of the visual optimization tools in {JPEG} 2000. *Signal Processing: Image Communication*, 17(1):85 – 104, 2002. {JPEG} 2000.

[256] Han Oh, A. Bilgin, and M.W. Marcellin. Visually lossless jpeg2000 using adaptive visibility thresholds and visual masking effects. In *Signals, Systems and Computers, 2009 Conference Record of the Forty-Third Asilomar Conference on*, pages 563–567, Nov 2009.

[257] D.L. McLaren and D.T. Nguyen. Removal of subjective redundancy from dct-coded images. *Communications, Speech and Vision, IEE Proceedings I*, 138(5):345–350, Oct 1991.

[258] Andrew B Watson. Dctune: A technique for visual optimization of dct quantization matrices for individual images. In *Sid International Symposium Digest of Technical Papers*, volume 24, pages 946–946. SOCIETY FOR INFORMATION DISPLAY, 1993.

[259] Gordon E. Legge. A power law for contrast discrimination. *Vision Research*, 21(4):457 – 467, 1981.

[260] Andrew B Watson. Dct quantization matrices visually optimized for individual images. In *IS&T/SPIE's Symposium on Electronic Imaging: Science and Technology*, pages 202–216. International Society for Optics and Photonics, 1993.

[261] Trac Duy Tran. *A locally adaptive perceptual masking threshold model for image coding*. PhD thesis, Massachusetts Institute of Technology. Dept. of Electrical Engineering and Computer Science, 1994.

[262] T.D. Tran and R. Safranek. A locally adaptive perceptual masking threshold model for image coding. In *Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on*, volume 4, pages 1882–1885 vol. 4, May 1996.

[263] Andrew B. Watson, Gloria Y. Yang, Joshua A. Solomon, and John D. Villasenor. Visual thresholds for wavelet quantization error, 1996.

[264] A.B. Watson, J.A. Solomon, and Jr. Ahumada, A.J. Visibility of dct basis functions: effects of display resolution. In *Data Compression Conference, 1994. DCC '94. Proceedings*, pages 371–379, Mar 1994.

[265] Siu-Wai Wu and A. Gersho. Rate-constrained picture-adaptive quantization for jpeg baseline coders. In *Acoustics, Speech, and Signal Processing, 1993. ICASSP-93., 1993 IEEE International Conference on*, volume 5, pages 389–392 vol.5, April 1993.

[266] W. C. Fong, S.C. Chan, and K.L. Ho. Designing jpeg quantization matrix using rate-distortion approach and human visual system model. In *Communications, 1997. ICC '97 Montreal, Towards the Knowledge Millennium. 1997 IEEE International Conference on*, volume 3, pages 1659–1663 vol.3, Jun 1997.

[267] M.G. Perkins and T. Lookabaugh. A psychophysically justified bit allocation algorithm for subband image coding systems. In *Acoustics, Speech, and Signal Processing, 1989. ICASSP-89., 1989 International Conference on*, pages 1815–1818 vol.3, May 1989.

[268] M.G. Perkins and T. Lookabaugh. A psychophysically justified bit allocation algorithm for subband image coding systems. In *Acoustics, Speech, and Signal Processing, 1989. ICASSP-89., 1989 International Conference on*, pages 1815–1818 vol.3, May 1989.

[269] I. Hontsch and L.J. Karam. Apic: adaptive perceptual image coding based on subband decomposition with locally adaptive perceptual weighting. In *Image Processing, 1997. Proceedings., International Conference on*, volume 1, pages 37–40 vol.1, Oct 1997.

[270] D Taubman. High performance scalable image compression with ebcot. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, volume Volume 3, pages 344–348, October 1999.

[271] David Taubman. High performance scalable image compression with ebcot. *Image Processing, IEEE transactions on*, 9(7):1158–1170, 2000.

[272] I. Hontsch and L.J. Karam. Locally adaptive perceptual image coding. *Image Processing, IEEE Transactions on*, 9(9):1472–1483, Sep 2000.

[273] Wenjun Zeng, S. Daly, and Shawmin Lei. Point-wise extended visual masking for jpeg-2000 image compression. In *Image Processing, 2000. Proceedings. 2000 International Conference on*, volume 1, pages 657–660 vol.1, 2000.

[274] Michael W Marcellin. *JPEG2000: image compression fundamentals, standards, and practice*, volume 1. springer, 2002.

[275] Peter Schelkens, Athanassios Skodras, and Touradj Ebrahimi, editors. *The JPEG 2000 Suite*. John Wiley & Sons, Chichester, UK, 2009.

[276] Sheila S. Hemami and Marcia G. Ramos. Wavelet coefficient quantization to produce equivalent visual distortions in complex stimuli, 2000.

[277] M.G. Ramos and S.S. Hemami. Perceptual quantization for wavelet-based image coding. In *Image Processing, 2000. Proceedings. 2000 International Conference on*, volume 1, pages 645–648 vol.1, 2000.

[278] Long-Wen Chang, Ching-Yang Wang, and Shiuh-Ming Lee. Designing jpeg quantization tables based on human visual system. In *Image Processing, 1999. ICIP 99. Proceedings. 1999 International Conference on*, volume 2, pages 376–380 vol.2, Oct 1999.

[279] J.R. Sullivan, L.A. Ray, and R. Miller. Design of minimum visual modulation halftone patterns. *Systems, Man and Cybernetics, IEEE Transactions on*, 21(1):33–38, Jan 1991.

[280] S. Daly. Subroutine for the generation of a two dimensional human visual contrast sensitivity function. Technical report 233203y, Eastman Kodak, Rochester, NY, 1987.

[281] A.C. Hung. Pvrg-jpeg codec. Tech. rep, Portable Video Research Group, Univ. Stanford, ftp://havefun.standard.edu/jpeg/JPEGv1.2.1.tar.Z, 1993.

[282] Zixiang Xiong, O.G. Guleryuz, and M.T. Orchard. A dct-based embedded image coder. *Signal Processing Letters, IEEE*, 3(11):289–290, Nov 1996.

[283] K. Ramchandran and M. Vetterli. Rate-distortion optimal fast thresholding with complete jpeg/mpeg decoder compatibility. *Image Processing, IEEE Transactions on*, 3(5):700–704, Sep 1994.

[284] M. Crouse and K. Ramchandran. Joint thresholding and quantizer selection for decoder-compatible baseline jpeg. In *Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on*, volume 4, pages 2331–2334 vol.4, May 1995.

[285] Marcus J. Nadenau and Julien Reichel. Compression of color images with wavelets under consideration of the hvs. In *in Proc. SPIE Human Vision and Electronic Imaging*, pages 129–140. SPIE, 1999.

[286] N. Damera-Venkata, T.D. Kite, W.S. Geisler, B.L. Evans, and A.C. Bovik. Image quality assessment based on a degradation model. *Image Processing, IEEE Transactions on*, 9(4):636–650, Apr 2000.

[287] G. Sreelekha and P.S. Sathidevi. An {HVS} based adaptive quantization scheme for the compression of color images. *Digital Signal Processing*, 20(4):1129 – 1149, 2010.

[288] Han Oh, A. Bilgin, and M.W. Marcellin. Visibility thresholds for quantization distortion in jpeg2000. In *Quality of Multimedia Experience, 2009. QoMEx 2009. International Workshop on*, pages 228–232, July 2009.

[289] Damon M Chandler, Nathan L Dykes, and Sheila S Hemami. Visually lossless compression of digitized radiographs based on contrast sensitivity and visual masking. In *Medical Imaging*, pages 359–372. International Society for Optics and Photonics, 2005.

[290] Han Oh, Ali Bilgin, and Michael W Marcellin. Visually lossless encoding for jpeg2000. *Image Processing, IEEE Transactions on*, 22(1):189–201, 2013.

[291] G. Bjontegaard. Calculation of average psnr differences between rd-curves (vceg-m33). Technical report, VCEG Meeting (ITU-T SG16 Q.6), Austin, Texas, USA, April 2001.

[292] Philippe Hanhart and Touradj Ebrahimi. Calculation of average coding efficiency based on subjective quality scores. *Journal of Visual Communication and Image Representation*, 25(3):555 – 564, 2014. QoE in 2D/3D Video Systems.

[293] S. Pateux. Tools for proposal evaluations. Technical Report JCTVC-A031, Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, Dresden, Germany, April 2010.

[294] Andersson K., Sjoberg R., and A Norkin. Reliability measure for bd measurements. Technical report, ITU-T SG16 Q.6 Document, VCEG-AL22, July 2009.

[295] A P. Beegan, L.R. Iyer, AE. Bell, V. R. Maher, and M. A Ross. Design and evaluation of perceptual masks for wavelet image compression. In *Digital Signal Processing Workshop, 2002 and the 2nd Signal Processing Education Workshop. Proceedings of 2002 IEEE 10th*, pages 88–93, Oct 2002.

[296] Otoniel M. Lopez, Miguel O. Martinez-Rach, Pablo Pi nol, Manuel Perez Malumbres, and JosÃ© Oliver. M-ltw: A fast and efficient intra video codec. *Signal Processing: Image Communication*, 23(8):637 – 648, 2008.

[297] Mathworks. Evaluating goodness of fit. http://www.mathworks.es/es/help/curvefit/evaluating-goodness-of-fit.html.

[298] Jinhua Yu. Advantages of uniform scalar dead-zone quantization in image coding system. In *Communications, Circuits and Systems, 2004. ICCCAS 2004. 2004 International Conference on*, volume 2, pages 805–808 Vol.2, June 2004.

[299] Michael W. Marcellin, Margaret A. Lepley, Ali Bilgin, Thomas J. Flohr, Troy T. Chinen, and James H. Kasner. An overview of quantization in {JPEG} 2000. *Signal Processing: Image Communication*, 17(1):73 – 84, 2002. {JPEG} 2000.

[300] Jang-Seon Ryu and Eung-Tea Kim. Fast intra coding method of h. 264 for video surveillance system. *Int. J. Comput. Sci. Netw. Secur*, 7(10):76–81, 2007.

[301] Michael Smith and John Villasenor. Intra-frame jpeg2000 vs. inter-frame compression comparison: The benefits and trade-offs for very high quality, high resolution sequences. *SMPTE Technical Conference and Exhibition, Pasadena, California*, pages 20–23, October 2004.

[302] Mourad Ouaret, Frederic Dufaux, and Touradj Ebrahimi. On comparing jpeg2000 and intraframe avc. In *SPIE Optics+ Photonics*, pages 63120U–63120U. International Society for Optics and Photonics, 2006.

[303] Pearson G. and Gill M.J. An evaluation of motion jpeg 2000 for video archiving. *Proc. Archiving*, (Washington, D.C.):237–243, April 2005.

[304] Boxin Shi, Lin Liu, and Chao Xu. Comparison between jpeg2000 and h.264 for digital cinema. In *Multimedia and Expo, 2008 IEEE International Conference on*, pages 725–728, June 2008.

[305] Jacob StrÃ¶m. Dead zone quantization in wavelet image compression - mini project in ece 253a, 1996.

[306] M. Martinez-Rach, O. Lopez, P. Piñol, J. Oliver, and M.P. Malumbres. A study of objective quality assessment metrics for video codec design and evaluation. In *Eight IEEE International Symposium on Multimedia*, volume 1, ISBN 0-7695-2746-9, pages 517–524, San Diego, California, Dec 2006. IEEE Computer Society.

[307] V. Galiano, O. Lopez, M.P. Malumbres, and H. Migallon. Parallel strategies for 2d discrete wavelet transform in shared memory systems and gpus. *The Journal of Supercomputing*, 64(1):4–16, 2013.

[308] Miguel O. Martinez-Rach, Otoniel Lopez-Granado, Vicente Galiano, Hector Migallon, Jesus Llor, and ManuelP Malumbres. Enhancing ltw

image encoder with perceptual coding and gpu-optimized 2d-dwt transform. *EURASIP Journal on Advances in Signal Processing*, 2013(1), 2013.