No-reference quality assessment of H.264/AVC encoded video

Tomás Brandão, *Student Member, IEEE,* and Maria Paula Queluz

***Abstract*—This paper proposes a no-reference quality assessment metric for digital video subject to H.264/AVC encoding. The proposed metric comprises two main steps: coding error estimation and perceptual weighting of this error. Error estimates are computed in the transform domain, assuming that DCT coefficients are corrupted by quantization noise. The DCT coefficient distributions are modeled using Cauchy or Laplace probability density functions, whose parameterization is performed using the quantized coefficient data and quantization steps. Parameter estimation is based on a maximum-likelihood estimation method combined with linear prediction. The linear prediction scheme takes advantage of the correlation between parameter values at neighbor DCT spatial frequencies. As for the perceptual weighting module, it is based on a spatio-temporal contrast sensitivity function applied to the DCT domain that compensates image plane movement by considering the movements of the human eye, namely smooth pursuit, natural drift and saccadic movements. The video related inputs for the perceptual model are the motion vectors and the frame rate, which are also extracted from the encoded video. Subjective video quality assessment tests have been carried out in order to validate the results of the metric. A set of eleven video sequences, spanning a wide range of content, have been encoded at different bitrates and the outcome was subject to quality evaluation. Results show that the quality scores computed by the proposed algorithm are well correlated with the mean opinion scores associated to the subjective assessment.**

***Index Terms*—Video quality, Image quality, No-reference metric, H.264, Parameter estimation.**

I. Introduction

OVER the last years, quality assessment of digital video has become an increasingly important matter, especially due to the transmission of video contents over the internet and mobile networks [1], [2]. Since human viewers are the target consumers for video communications products, they are the most reliable source for assessing their quality. However, gathering video quality assessment data from the human viewers is not an easy task, since it requires the completion of *subjective quality tests*. A standardization of the procedures for conducting these tests is described in ITU recommendations [3], [4] and the quality scores that result from such experiments are usually addressed to as *subjective scores* or *mean opinion scores* (MOS). Subjective tests must be carried out in a controlled environment and they require quality judgments performed by several viewers. Thus, subjective quality scores are hard to get and they cannot be used in real-time applications.

T. Brandão is with the Department of Technology and Information Sciences, ISCTE-Lisbon University Institute, and with Instituto de Telecomunicações, Lisbon; e-mail: tomas.brandao@lx.it.pt.

M. P. Queluz is with the Department of Electrical and Computer Engineering, IST-Technical University of Lisbon, and with Instituto de Telecomunicações, Lisbon; e-mail: paula.queluz@lx.it.pt

An alternative to subjective quality assessment is to automatically score video quality using *objective metrics*. Most of the research performed in this field has been focused on the development of *full reference* (FR) metrics (some examples in [5]–[9]), which require both the original and the distorted video data to compute the quality scores. FR metrics are typically used for benchmarking image and video processing algorithms, such as lossy encoding or watermarking techniques, and media distribution networks during the testing phases. However, FR metrics are not suitable for monitoring the quality of received media once the distribution network is setup and starts working, since the original data is usually not available at the receiver.

It is thus desirable to have a quality measurement system at the receivers that is able to provide quality feedback without requiring the reference signals. This has led to an increased research effort on *no-reference* (NR) quality metrics [10]–[17] and *reduced reference* (RR) quality metrics [18]–[21]. NR metrics rely on the received media only. RR metrics can be placed between FR and NR metrics: information about the reference is sent through a side information channel and is used at the receiver for computing the objective quality scores. RR and NR quality metrics for video may contribute to enabling new services and applications, such as *quality of experience* (QoE) monitoring, scalable billing schemes, and real-time adjustment of streaming parameters as a function of the perceived quality.

The method proposed in this paper assesses video quality without requiring any knowledge about the original signal, thus belonging to the NR quality metrics class. In short, it consists of local error estimation followed by perceptual spatio-temporal error weighting and pooling.

Error estimates rely on statistical properties of the block-based *discrete cosine transform* (DCT) coefficient data. Since the proposed metric belongs to the no-reference class, it is necessary to accurately estimate the distribution of the original DCT coefficients using the received (corrupted) coefficient data. Related work on video coding error estimation is presented in [14]–[17]. In [14], Turaga *et al.* were probably the first authors to propose a no-reference image quality assessment algorithm that estimates video *peak signal-to-noise ratio* (PSNR) based on the statistical properties of DCT coefficients. Their work is oriented to MPEG-2 encoded video and the statistical distribution of the DCT coefficients are modeled using Laplace *probability density functions* (PDFs). However, as the number of DCT coefficients quantized to zero values increases, the estimation of the Laplace density parameter becomes inaccurate. Aware of this situation, Ichigaya *et al.* proposed in [15] an improvement for the DCT coefficient

distribution model by using a weighted mixture of Laplacian PDFs: one is computed by considering all quantized coefficient values and the other is computed by considering the non-zero quantized values only. However, the method in [15] still fails when all DCT coefficients at the same frequency are quantized to zero.

In a more recent work [16], Eden proposes a PSNR estimation method for H.264 encoded video sequences. The coefficients' distributions are modeled according to Laplace densities, using a low complexity algorithm for the estimation of the density's parameter, tackling the "all coefficients quantized to zero" problem by imposing bounds in the parameter's value at the corresponding frequencies. The results depicted in [16] show that this strategy provides good PSNR estimates for I-frames but the results for P and B-frames still need to be improved.

All the above mentioned works estimate PSNR values, which are known to be not well correlated with the human perception of quality [22]. In [23], the authors propose a no-reference quality assessment method for still images subject to JPEG encoding that, besides producing PSNR estimates, also outputs MOS estimates that are proven to be well correlated with the corresponding subjective assessment data. This method also estimates local errors in the DCT coefficient's domain, but weights those errors perceptually, using the *just noticeable difference* (JND) perceptual model proposed by Watson in [24].

This paper generalizes the method proposed in [23] to the more challenging case of encoded video sequences. Although the H.264 standard and its corresponding integer DCT [25], [26] have been considered, the method can be straightly applied to any DCT-based video encoding scheme. It starts by estimating the DCT coefficient's error, assuming that these are corrupted by quantization noise only. Error estimates that result from this procedure are then perceptually weighted, by considering characteristics of the human eye, namely its sensitivity to spatio-temporal contrast. A spatio-temporal perceptual model based on the work of Kelly and Daly is used. In [27], Kelly devised an analytic model for the spatio-temporal CSF, based on data collected from his experiments. His work was further extended by Daly in [28] by considering movements of the eye, namely *smooth pursuit*, *natural drift* and *saccadic* eye movements.

In order to evaluate the results of the metric derived in this paper, a set of subjective tests have been conducted. The methodology followed in these tests is in accordance with Recommendation ITU-T P.910 [4].

This paper is organized as follows: in section II, the no-reference quality estimation framework is introduced and its modules are detailed in sections III and IV. Results and a short description of subjective tests are depicted in section V. The main conclusions and topics for further research are given in section VI.

II. No-reference quality assessment framework

The proposed framework for assessing the quality of an H.264 encoded video sequence is represented in figure 1. It consists of two main blocks: an error estimation block, whose function is to compute local error estimates, and a perceptual weighting block, whose function is to weight and combine those error estimates, in order to compute a quality score. This type of architecture allows to extend the use of perceptual models (based on error weighting) to the no-reference quality assessment problem.

Suppose that the distribution of the original DCT coefficient data is known. In this case, an estimate for the local mean square error, $\hat{\varepsilon}_k^2$, at the k-th coefficient, can be performed by observing the value of its quantized value, X_k:

$$\hat{\varepsilon}_k^2 = \int_{-\infty}^{+\infty} f_X(x|X_k)(X_k - x)^2 dx, \tag{1}$$

where $f_X(x|X_k)$ represents the distribution of the original DCT coefficients values conditioned to the observed value of X_k. Using *Bayes rule* for conditional densities [29] and considering that $P(X_k|x) = 1$ if x is in the quantization interval around X_k, $[a_k; b_k]$, and $P(X_k|x) = 0$, otherwise, (1) can be rewritten as:

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

where $f_X(x)$ is the original coefficient data distribution and the quantization interval limits a_k and b_k are defined as [25]:

$$\begin{cases} a_k = -\alpha q_k \\ b_k = \alpha q_k, \end{cases} \quad \text{if } X_k = 0; \qquad \begin{cases} a_k = |X_k| - (1-\alpha) q_k \\ b_k = |X_k| + \alpha q_k, \end{cases} \quad \text{if } X_k \neq 0, \tag{3}$$

where q_k is the quantization step and α is a parameter that controls the width of the quantizer's dead zone around 0. In the reference H.264 software [30], $\alpha \simeq 2/3$ for intra blocks and $\alpha \simeq 5/6$ for inter blocks. The quantization step, q_k, can be derived from a bitstream parameter called QP, which may differ from macroblock to macroblock [25].

From (2), it can be concluded that the squared error estimate depends on the value of the quantized coefficient X_k, on the quantization step q_k (which determines a_k and b_k) and on the coefficient distribution $f_X(x)$. X_k and q_k can be derived from the encoded bitstream. As for $f_X(x)$, it is estimated from the available quantized data, as will be explained in section III.

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

$$\mathrm{PSNR}_{\mathrm{est}_{[\mathrm{dB}]}} = 10 \log_{10} \frac{255^2}{\mathrm{MSE}_{\mathrm{est}}}; \quad \mathrm{MSE}_{\mathrm{est}} = \frac{1}{N} \sum_{k=1}^{N} \hat{\varepsilon}_k^2, \tag{4}$$

where N is the number of DCT coefficients. Note that, in accordance with Parceval's theorem, it is indifferent to measure the PSNR in the pixel or in the DCT domain. The DCT coefficient error estimates are then perceptually weighted using a spatio-temporal perceptual model based on [27], [28]. The function of this model is to compute local perceptual weights p_k, which reflect the sensibility of the HVS to the

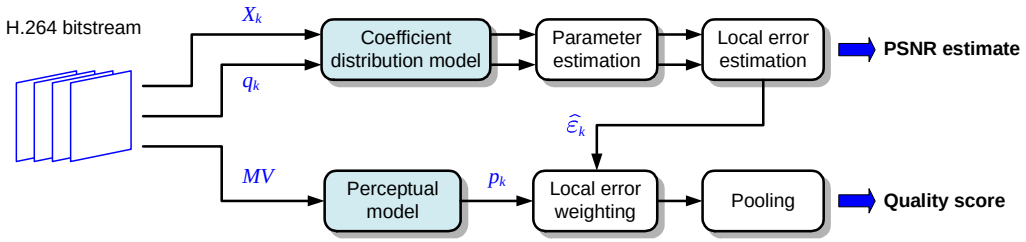


Fig. 1. Architecture of the proposed quality assessment metric.

corresponding local errors. The inputs for the model are the motion vectors, MV, and the video frame rate, f_r, both extracted from the encoded bitstream. From the weighted local errors, $p_k\hat{\varepsilon}_k$, a global perceptual distortion metric is obtained using error pooling.

III. Modeling DCT coefficient data

Block-based DCT coefficient data distribution of natural images are typically modeled by zero-mean *Laplace* [31] or *Cauchy* [32], [33] PDFs. Other DCT coefficient distribution models have been suggest in the literature, such as generalized gaussian [34], gaussian mixtures [35], or generalized gamma [36]. The support for those distribution models consisted of theoretical (i.e., the *Central Limit* theorem) and quantitative results (i.e., the χ-square and Kolmogorov-Smirnov goodness-of-fit tests). On this work, the aforementioned zero-mean *Laplace* and *Cauchy* models have been considered. Both models require the estimation of a single parameter and represent a reasonable trade-off between accuracy and simplicity.

In the following, the methodology for estimating the distribution's parameter is described for both models, using the original and the quantized (corrupted) DCT coefficient data.

A. Cauchy model

Using $K \times K$ DCT blocks, for each horizontal/vertical frequency pair, $(i,j) \in \{0,...,K-1\} \times \{0,...,K-1\}$, the coefficient's distribution is modeled by:

$$f_X(x)_{(i,j)} = \frac{1}{\pi}\frac{\beta_{(i,j)}}{\beta_{(i,j)}^2 + x^2}, \tag{5}$$

where $\beta_{(i,j)}$ is the distribution's parameter and x represents the coefficient's value at spatial frequency (i,j). For simplicity, the indexes (i,j) will be dropped along the text, but it must be kept in mind that there is a distinct parameter value at each spatial frequency.

1) Estimating β using the original coefficient values: If the original coefficient data is known, an estimate for parameter β can be computed using the *maximum-likelihood* (ML) method [29]:

$$\beta_{ML} = \arg\max_{\beta}\left\{\log\prod_{k=1}^{N} f_X(x_k)\right\}, \tag{6}$$

where x_k is the k-th coefficient value and N is the number of coefficients at the frequency under analysis. Using (5) in (6) leads to:

$$\beta_{ML} = \arg\max_{\beta}\left\{\sum_{k=1}^{N}\left(\log\beta - \log(\beta^2 + x_k^2)\right)\right\}. \tag{7}$$

The value of β that maximizes (7) can be computed by finding the zeros of the derivative with respect to β, which leads to:

$$\frac{N}{\beta} - 2\sum_{k=1}^{N}\frac{\beta}{\beta^2 + x_k^2} = 0. \tag{8}$$

To solve (8), *Newton-Raphson*'s root finding method was used, starting with a small value (0.1) as the initial value for β. Convergence has been achieved in all experiments. The resulting value for β_{ML} can be seen as a reference value, thus it will be addressed to as the "original" parameter value.

2) Estimating β using quantized coefficient values: Now, let's suppose that only quantized data is available for estimating β, which is the case at the receiver (decoder) side. The ML method can still be used:

$$\hat{\beta}_{ML} = \arg\max_{\beta}\left\{\log\prod_{k=1}^{N} P(X_k)\right\}, \tag{9}$$

where $P(X_k)$ represents the probability of having value X_k at the quantizer's output. Assuming that the quantizer is linear with step size q_k, which may differ from block to block, and that it includes a dead zone around 0, controlled by parameter α, $P(X_k)$ can be written as:

$$\begin{aligned} P(X_k) &= \int_{a_k}^{b_k}\frac{1}{\pi}\frac{\beta}{\beta^2+x^2}dx \\ &= \begin{cases} \frac{2}{\pi}\tan^{-1}\left(\frac{\alpha q_k}{\beta}\right), & \text{if } X_k = 0; \\ \frac{1}{\pi}\left(\tan^{-1}\left(\frac{b_k}{\beta}\right) - \tan^{-1}\left(\frac{a_k}{\beta}\right)\right), & \text{otherwise.} \end{cases} \end{aligned} \tag{10}$$

Using (10) in (9) leads to:

$$\begin{aligned} \hat{\beta}_{ML} = \arg\max_{\beta}\Bigg\{ &\sum_{k_0=1}^{N_0}\log\left(\frac{2}{\pi}\tan^{-1}\left(\frac{\alpha q_{k_0}}{\beta}\right)\right) + \\ &+ \sum_{k_1=1}^{N_1}\log\frac{1}{\pi}\left(\tan^{-1}\left(\frac{b_{k_1}}{\beta}\right) - \tan^{-1}\left(\frac{a_{k_1}}{\beta}\right)\right)\Bigg\} \end{aligned} \tag{11}$$

The two summation terms in (11) correspond to the two possible cases in (10). In practice, the set of quantized coefficients X_k has been split according to those cases: quantized

coefficients with zero and non-zero values, respectively. Accordingly, N_0 and N_1 represent the number of coefficients (at a given frequency), that fall in those cases. The value of β that maximizes (11) can be obtained by finding the zero of the derivative with respect to β, which corresponds to:

$$\sum_{k_1=1}^{N_1} \frac{\frac{a_{k_1}}{\beta^2+a_{k_1}^2} - \frac{b_{k_1}}{\beta^2+b_{k_1}^2}}{\tan^{-1}\left(\frac{b_{k_1}}{\beta}\right) - \tan^{-1}\left(\frac{a_{k_1}}{\beta}\right)} - \\ - \sum_{k_0=1}^{N_0} \frac{\alpha q_{k_0}}{\tan^{-1}\left(\frac{\alpha q_{k_0}}{\beta}\right)\left((\alpha q_{k_0})^2+\beta^2\right)} = 0. \tag{12}$$

If $N_0 < N$, a solution for (12) can be found numerically, using the same method as in (8). If $N_0 = N$, then $\beta \to 0$, meaning that the estimated coefficient distribution is a *Dirac's delta* function centered in 0. In other words, the ML method will fail if all coefficients at a given frequency are quantized to zero.

B. Laplace model

Using $K \times K$ blocks, for each horizontal/vertical frequency pair, $(i, j) \in \{0, ..., K-1\} \times \{0, ..., K-1\}$, the coefficient's distribution for the Laplace model is described by:

$$f_X(x) = \frac{\lambda}{2} \exp(-\lambda|x|), \tag{13}$$

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

1) Estimating λ using the original coefficient values: Following a procedure similar to what has been done in section III-A, an ML estimation for λ, using the original coefficient data, is given by:

$$\lambda_{ML} = \arg\max_{\lambda} \left\{ \sum_{k=1}^{N} \left(\log\left(\frac{\lambda}{2}\right) - \lambda|x_k| \right) \right\} \tag{14}$$

where N represents the number of coefficients at the given frequency and x_k is the k-th coefficient value at that frequency. Differentiating the function inside $\arg\max\{.\}$ with respect to λ, and finding the zeros, will lead to:

$$\lambda_{ML} = \frac{N}{\sum_{k=1}^{N} |x_k|}, \tag{15}$$

a result that is well-known from literature [37].

2) Estimating λ using quantized coefficient values: Assuming that only quantized data is available for parameter estimation, λ can be computed using the ML method in the same way as in (9). For this case, the probability $P(X_k)$ can be written as:

$$P(X_k) = \int_{a_k}^{b_k} \frac{\lambda}{2} \exp(-\lambda|x|)dx \\ = \begin{cases} 1 - e^{-\lambda b_k}, & \text{if } X_k = 0; \\ \frac{1}{2} e^{-\lambda b_k}(e^{\lambda q_k} - 1), & \text{otherwise.} \end{cases} \tag{16}$$

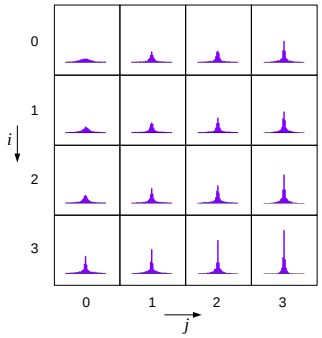


Fig. 2. Typical coefficient histograms on I-frames under H.264 encoding (original coefficient values taken from an I-frame of sequence *Stephan*).

Using (9) for the laplacian and substituting $P(X_k)$ by the result in (16) leads to:

$$\hat{\lambda}_{ML} = \arg\max_{\lambda} \left\{ \sum_{k_0=1}^{N_0} \log(1 - e^{-\lambda b_{k_0}}) + \right. \\ \left. + \sum_{k_1=1}^{N_1} \log(e^{\lambda q_{k_1}} - 1) - \lambda b_{k_1} \right\} \tag{17}$$

Again, the value that maximizes (17) can be found by looking for the zeros of the derivative with respect to λ, which leads to:

$$\sum_{k_0=1}^{N_0} \frac{b_{k_0}}{e^{\lambda b_{k_0}} - 1} + \sum_{k_1=1}^{N_1} \left(\frac{q_{k_1} e^{\lambda q_{k_1}}}{e^{\lambda q_{k_1}} - 1} - b_{k_1} \right) = 0. \tag{18}$$

Once more, the solution can be found by using an iterative root finding algorithm. However, if all coefficients have been quantized to zero, *i.e.* $N = N_0$, only the first sum term of (18) stands, leading to:

$$\sum_{k=1}^{N} \frac{b_k}{e^{\lambda b_k} - 1} = 0 \tag{19}$$

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

C. Improving estimation using prediction

In order to enable PDF parameter estimation at the frequencies where all DCT coefficients were quantized to zero, as described at the end of sections III-A2 and III-B2, the correlation between parameter values at neighboring DCT frequencies can be explored. Consider figure 2, which represents a set of histograms for the H.264 coefficient values, one histogram per spatial frequency, in a given test frame. As can be observed from the plots, as frequency increases the histogram shape becomes increasingly narrow (which means that the variance of the coefficient values decreases as frequency increases).

Additionally, figure 3 depicts the "original" β and λ values, computed using equations (12) and (15), of a test I-frame subject to H.264 encoding. The figures show that there is a strong correlation between parameter values at adjacent

frequencies. Although these plots are related to a particular example, a similar evolution is verified on other I frames, and also in P and B frames. The plots also show that a similar evolution is verified in both possible H.264 transform sizes (4×4 and 8×8). In order to support these statements, the correlation between neighboring parameter values in a 4-connected neighborhood has been measured considering all the frames used in the experiments (see section V). For the 4×4 sized transform, those measurements were of 0.92, 0.91 and 0.93 for I, P and B frames, respectively.

One possible way to explore this correlation, is to use a linear predictor, as suggested in [23] for still images. Representing the predicted parameter value by $\hat{\theta}_p$, where θ can either be the Cauchy's β or the Laplace's λ, it can be written:

$$\hat{\theta}_p = \boldsymbol{\theta}^{\mathrm{T}}\mathbf{w}, \tag{20}$$

with

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

where K_v is the neighborhood size, θ_k is the parameter value at the k-th neighbor and w_k is the associated linear weight.

The prediction value, $\hat{\theta}_p$, that results from (20) is combined with the parameter's ML estimate, $\hat{\theta}_{ML}$, in order to improve the estimation accuracy for the DCT distribution's parameter. Since ML estimates become more inaccurate as the rate of coefficients quantized to zero increases, more trust should be given to the predictor in these situations. On the other hand, if the number of coefficients quantized to zero is low, the ML estimator will most likely get accurate results, so there is no real need for the predicted value. Based on these premises, a simple criterion for combining $\hat{\theta}_p$ with $\hat{\theta}_{ML}$ is to weight them according to:

$$\hat{\theta}_f = {r_0}^{\gamma}\hat{\theta}_p + (1 - {r_0}^{\gamma})\hat{\theta}_{ML}, \tag{21}$$

where $\hat{\theta}_f$ is the final estimation for the distribution's parameter, $r_0 = \frac{N_0}{N}$ represents the rate of coefficients quantized to zero and the exponent γ regulates how fast the confidence on the ML estimates decrease with increasing r_0. The best results were obtained using $\gamma = 2$.

D. Predictor training

The goal of the training procedure is to find a weight vector $\mathbf{w}$ suitable for the linear prediction scheme given in (20). One possible way is to compute $\mathbf{w}$ by minimizing the square error between the "original" and predicted parameter values in a given training set, subject to a penalty on the size of the linear weights, in a procedure known as *Ridge regression* [38]. According to this method, the linear weights can be found by solving:

$$\hat{\mathbf{w}}_{ridge} = \arg\min_{\mathbf{w}} \left\{ \sum_{i=1}^{N} (\theta_i - \hat{\theta}_i)^2 + \alpha \sum_{k=1}^{K_v} w_k^2 \right\}, \tag{22}$$

where N is the number of video frames available for training, K_v is the neighborhood size and α is a positive value that controls the penalty applied to the value of the weights (note that, for $\alpha = 0$, this method falls in the pure least squares solution). Since there are N video frames, there will also be N "original" parameter values of θ and their corresponding neighborhood vectors $\boldsymbol{\theta}$ per frequency. Using matrix notation, (22) can be rewritten as:

$$\hat{\mathbf{w}}_{ridge} = \arg\min_{\mathbf{w}} \left\{ (\boldsymbol{\theta} - \boldsymbol{\Theta}\mathbf{w})^{\mathrm{T}}(\boldsymbol{\theta} - \boldsymbol{\Theta}\mathbf{w}) + \alpha \mathbf{w}^{\mathrm{T}}\mathbf{w} \right\}, \tag{23}$$

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

$$\boldsymbol{\Theta} = \begin{bmatrix} \theta_{11} & \dots & \theta_{1K_v} \\ \theta_{21} & \dots & \theta_{2K_v} \\ \vdots & & \vdots \\ \theta_{N1} & \dots & \theta_{NK_v} \end{bmatrix}, \boldsymbol{\theta} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_N \end{bmatrix}. \tag{24}$$

The solution that minimizes (23) can be found by differentiating with respect to $\mathbf{w}$:

$$\nabla_{\mathbf{w}} = 0 \Leftrightarrow -2\boldsymbol{\Theta}^{\mathrm{T}}(\theta - \boldsymbol{\Theta}\mathbf{w}) + 2\alpha\mathbf{w} = 0, \tag{25}$$

leading to

$$\hat{\mathbf{w}}_{ridge} = (\boldsymbol{\Theta}^{\mathrm{T}}\boldsymbol{\Theta} + \alpha\mathbf{I})^{-1}\boldsymbol{\Theta}^{\mathrm{T}}\boldsymbol{\theta}. \tag{26}$$

The neighborhood configuration used by the error estimation module is illustrated in figure 4. Since low-frequency coefficients are less vulnerable to the effects of lossy encoding, its structure has been chosen with the purpose of predicting parameter values based on predictions already performed at lower frequencies.

The training procedure can be synthesized in the following steps:

1) for each original image in the training set, compute the "original" parameter values with (8), if using Cauchy model, or with (15), if using Laplace model;
2) for each encoded video frame in the training set, compute r_0 and $\hat{\theta}_{ML}$ using (12) or (18) for all spatial frequencies;
3) for each DCT frequency, in zig-zag scan order:
   a) build the neighborhood matrix $\boldsymbol{\Theta}$. The values of $\hat{\theta}_f$ are computed using the values of r_0 and $\hat{\theta}_{ML}$ that result from step 2, as well as previously computed predictions (if not computed yet, assume that $\hat{\theta}_f = \hat{\theta}_{ML}$)
   b) build $\boldsymbol{\theta}$ using the values that result from step 1;
   c) compute the weight vector $\mathbf{w}$ for the current frequency position, using (26);
   d) use the resulting values of $\mathbf{w}$ to perform predictions at that frequency (which will be used in step (a) in posterior iterations).

IV. Perceptual model

The function of the perceptual model is to weight and combine the local error estimates that result from the module described in the previous section. It is based on the CSF

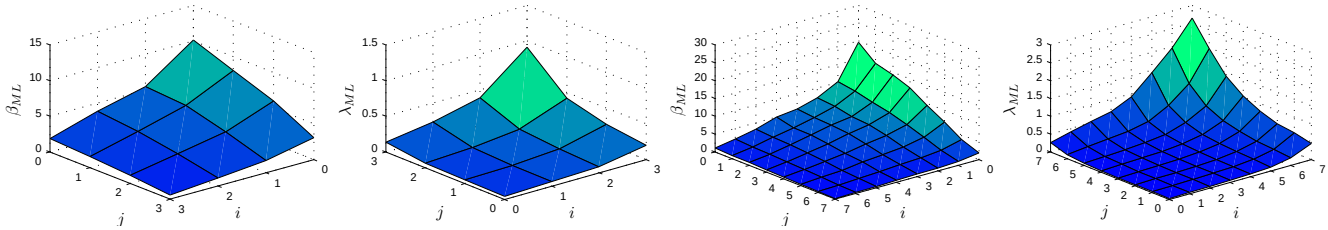


Fig. 3. Typical evolution of the H.264 coefficient's distribution parameter as a function of the spatial frequency (original coefficient values taken from an I-frame of sequence *Stephan*). (a) β parameter (Cauchy) – 4×4 transform. (b) λ parameter (Laplace) – 4×4 transform. (c) β parameter (Cauchy) – 8×8 transform. (d) λ parameter (Laplace) – 8×8 transform.

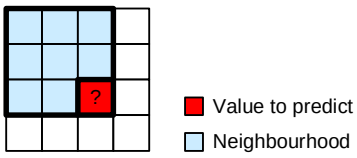


Fig. 4. Neighborhood configuration used in the experiments.

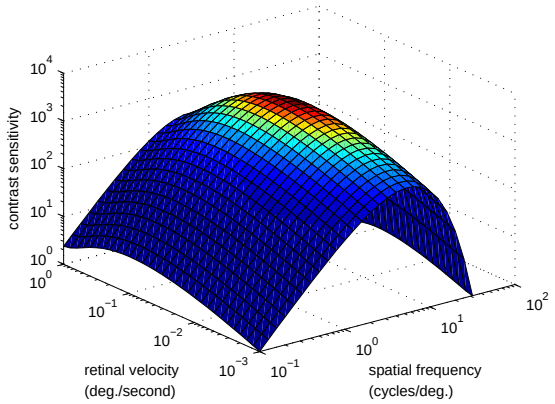


Fig. 5. Spatio-temporal contrast sensitivity function (based in the model by *Daly* [28]).

derived in [27] and extended in [28], accounting for the mechanics of the human eye. Since the goal of the metric proposed in the paper is to perform no-reference video quality assessment, only video elements available at the decoder are used: the motion vectors, MV, and the video frame rate, f_r.

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

A. Spatio-temporal CSF model

It his known that the *human visual system* (HVS) is more sensitive to image contrast rather than the absolute luminance values [1], [39]. Contrast can be defined as the ratio between the local luminance variation and the average background luminance. *Contrast sensitivity* is the inverse of the minimum contrast necessary for an observer to detect a stimulus. A spatio-temporal *Contrast sensitivity function* (CSF) quantifies the evolution of the HVS sensitivity to luminance changes and depends on the spatial and temporal frequencies of the stimulus. Different CSFs have been proposed in literature [27], [28], [40], [41]. In the model by Kelly [27] and Daly [28], the spatio-temporal sensitivity is computed as a function of the spatial frequency, f_s, and the retinal velocity, v_R, as follows:

$$CSF(v_R, f_s) = Sc_0c_2v_R(2\pi c_1 f_s)^2 \exp\left(-\frac{4\pi c_1 f_s}{f_{max}}\right), \quad (27)$$

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

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

The constants s_1, s_2 and p_1 have been set to 6.1, 7.3 and 45.9, respectively [27]. The parameters c_0, c_1 and c_2 allow model tunning and have been set to the same values as in [28]: $c_0 = 1.14$, $c_1 = 0.67$ and $c_2 = 1.7$. Figure 5 depicts the resulting CSF.

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

$$f_s = \sqrt{f_x^2 + f_y^2}. \quad (28)$$

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

$$f_y = \frac{i}{2K\alpha_y} \text{ and } f_x = \frac{j}{2K\alpha_x}, \quad (29)$$

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

$$\alpha_\phi = \arctan\frac{l_\phi}{2dN_\phi} \simeq \frac{l_\phi}{2dN_\phi}. \quad (30)$$

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

The object velocity on the retina plane is strongly related with the object velocity in the image plane. However, the human eye has the ability to track objects, slowing down the velocity of the object in the retina plane. This characteristic is called the *smooth pursuit eye movement* (SPEM). Additionally, there are other movements of the eye, namely the *natural drift* and *saccadic* eye movements [42]. The former is a slow eye movement that causes a little amount of motion in the

retina plane, while the latter are fast eye movements caused by changing the eye gaze to new image plane locations.

According to [28], the retinal image velocity can be computed as:

$$v_R = v_I - v_E, \tag{31}$$

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

$$v_E = \min\{g_S \times v_I + v_{MIN}; v_{MAX}\}, \tag{32}$$

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

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

$$v_I = f_r \sqrt{(MV_x \alpha_x)^2 + (MV_y \alpha_y)^2}, \tag{33}$$

where f_r is the frame rate of the video sequence and (MV_x, MV_y) are the components of the motion vector along the horizontal and vertical directions, respectively. The components of the observation angle of a pixel, α_x and α_y, are those resulting from (29).

B. Quality scores

Based on the result of the CSF compute at each location of the block-wise DCT domain, a global distortion value for each video frame, D_f is computed using $L4$ error pooling, as suggest in [9], [43], according to:

$$D_f = \sqrt[4]{\sum_k (\hat{\varepsilon}_k p_k)^4}, \tag{34}$$

where $p_k = \text{CSF}(v_{r_k}, f_{s_k})$ is the result of the contrast sensitivity function at the k-th DCT coefficient's location and $\hat{\varepsilon}_k$ is the error estimate that results from the error estimation module. The use of $L4$ error pooling emphasizes higher distortions perceived by the viewer, which may drawn his visual attention from smaller distortions. To conclude, the same pooling process is applied along the time axis in order to get a global distortion metric for the encoded video sequence:

$$D_g = \sqrt[4]{\sum_i {D_{f_i}}^4}. \tag{35}$$

Note that, for longer video sequences, a granularity period for computing D_g could be defined (*e.g.*, D_g could be computed every 10 seconds of video).

V. Results

A. Subjective quality assessment

In order to validate the results of the proposed metric, a set of subjective quality assessment tests have been carried out. Those tests performed in accordance with the *Degradation Category Rating* (DCR) described in Recommendation ITU-T P.910 [4], where video sequences are presented in pairs: the first to be displayed is called the *reference* sequence (in our case, the original sequence) while the second is called the *test* or *impaired* sequence (in our case, the result of lossy encoding). The observers are then asked to judge the quality of the impaired sequence with respect to the reference. A five point impairment scale has been used, with grades from "1 – very annoying" to "5 – imperceptible". 42 observers (mostly students) participated in the subjective experiments. They were screened for visual acuity and color blindness. The environmental viewing conditions were within the values recommended in [4].

The reference sequences used in the tests are represented in figure 6. These sequences are in CIF format (352×288), with a frame rate of 30 Hz, and have been selected in order to span a wide range of spatio-temporal activities. The sequences were encoded using the reference H.264 [30] software tools. Each sequence has been encoded using the main profile at different bit rates, which were in the range from 32 to 2048 kbit/s. A GOP-15 structure with two B frames inserted between I/P frames ($IBBPBBP...$) has been used in all encoding runs. The low complexity rate-distortion optimization algorithm provided on the software has been used. The result is a set of 50 encoded sequences (impaired sequences), whose qualities were judged by the test participants.

The resulting MOS values and the video sequences used in the subjective quality assessment tests, as well as additional details about the test procedures, are available online at http://amalia.img.lx.it.pt/~tgsb/H264_test/.

B. Prediction accuracy

The training of the parameter prediction module was performed using about one third of the available video samples, following the procedure described in section III-D. Training has been performed separately for each frame type. Based on the results presented in [17], the Cauchy model was assigned to the I-frames, while the Laplace model was assigned to the P and B frames.

The effectiveness of the proposed prediction scheme has been evaluated using the remaining samples. To illustrate the results, table I-a) presents the root mean square (RMS) error between "original" and ML estimated parameter values for the I-frames (previously normalized to zero mean and unitary variance). Similarly, table I-b) presents the RMS between "original" and prediction estimates alone. As for table I-c), it presents the error that results from combining prediction with ML estimates. It can be observed that the improvement brought by the prediction scheme becomes more noticeable as frequency increases. For the low frequency coefficients, the number of neighborhood elements is small, thus the improvement brought by using prediction is not as effective as for the high frequency coefficients.

In addition, figure 7 depicts an example that illustrates the estimation of the Cauchy parameter, β, in the presence of H.264 encoding. Figure 7-a) shows the "original" values of β that result from solving (6), which can be seen as the no-reference estimation benchmark. Figure 7-b) shows the results of ML parameter estimation based on the quantized data. As can be observed from this plot, the parameter could not be estimated at seven spatial frequencies, due to all DCT

Fig. 6. Video sequences selected for the subjective tests. From left to right, up to down: *City*; *Coastguard*; *Container*; *Crew*; *Football*; *Foreman*; *Mobile & Calendar*; *Silent*; *Stephan*; *Table-tennis*; *Tempete*.

TABLE I
PARAMETER ESTIMATION ERROR.

$i\downarrow$ \ $\underrightarrow{j}$				
	0.08	0.66	0.92	1.81
	0.44	1.20	1.38	2.15
	0.87	2.02	2.06	2.73
	2.99	3.96	3.69	4.72

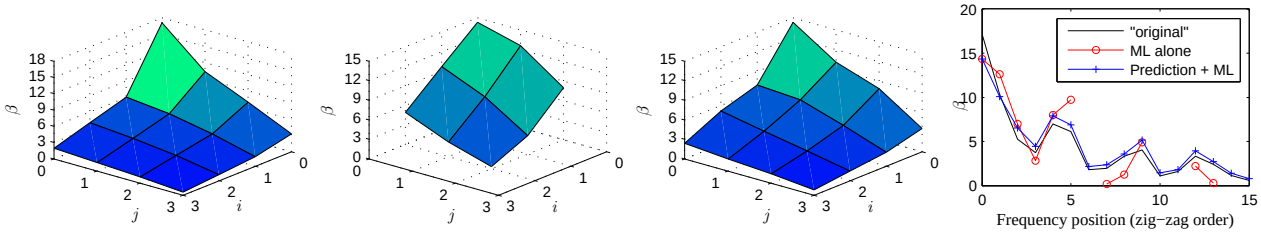
(a) ML estimates alone.

$i\downarrow$ \ $\underrightarrow{j}$				
	0.08	0.44	0.30	0.30
	0.42	0.26	0.28	0.28
	0.38	0.30	0.29	0.27
	0.38	0.46	0.38	0.32

(b) Prediction estimates.

$i\downarrow$ \ $\underrightarrow{j}$				
	0.08	0.25	0.29	0.30
	0.28	0.26	0.28	0.29
	0.37	0.32	0.30	0.27
	0.40	0.45	0.38	0.32

(c) ML combined with prediction.



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

Fig. 7. Example of parameter estimation (on a H.264 encoded I-Frame, using Cauchy model). (a) "Original" values. (b) ML estimates alone. (c) Combining ML with prediction. (d) Comparison.

coefficients quantized to zero at those frequencies. After using the predictor, the missing parameter values are computed and the estimates are improved, as shown in figure 7-c). For a better comparison, figure 7-d) depicts in a 2D plot the information of the previous plots.

Note that, since all video sequences were encoded using the H.264's main profile, the results and corresponding plots were obtained for the 4×4 sized transform size only. Nevertheless, and considering the plots depicted in figures 3-c) and d), a similar process is expected to work in higher H.264 profiles, where the 8×8 transform is allowed. In such cases, distribution parameter predictors should be trained separately for each transform size.

C. PSNR estimation

Using the full set of encoded video sequences, the PSNR has been estimated and compared with its true value. Results are depicted in figures 8-a) to d), for the different frame types. As can be observed from the plots, the proposed method is quite accurate. Note that an additional procedure has been performed in order to compensate for the occurrence of *skipped* macroblocks, which become quite common in P and B frames as the encoding bit rate decreases. This compensation procedure is given by:

$$\mathrm{MSE}_{\mathrm{est}} = r_s \times \mathrm{MSE}_{\mathrm{ref}} + (1 - r_s) \times \mathrm{MSE}_{\varepsilon}, \qquad (36)$$

where r_s is the rate of skipped MBs within the frame under analysis, $\mathrm{MSE}_{\mathrm{ref}}$ is the MSE of the reference frame(s) and $\mathrm{MSE}_{\varepsilon}$ is the mean square error estimate computed by the algorithm, considering the nonskipped MBs only.

For comparison purposes, the algorithm proposed by *Eden* in [16] has been implemented. This algorithm models coefficient distribution using a Laplace PDF, and uses a low complexity parameter estimation method for computing λ, which is given by:

$$\hat{\lambda}_{Eden} = -\frac{\log(1 - r_0)}{\alpha \bar{q}}, \qquad (37)$$

where $\bar{q}$ is the average quantization step used at a given DCT frequency within one frame and the remaining parameters are

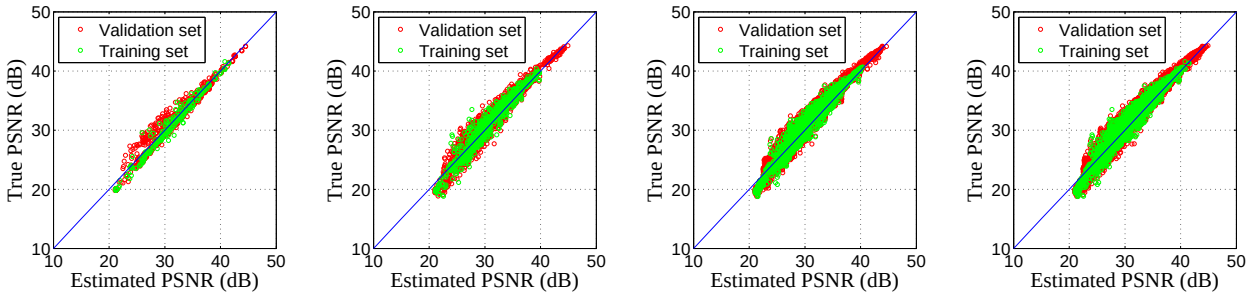


Fig. 8. No-reference PSNR estimation *vs.* true PSNR. (a) I-frames. (b) P-frames. (c) B-frames. (d) All frames.

TABLE II
PSNR ESTIMATION ERROR.

	Eden's [16]			Proposed		
Frame Type	ε_{avg}	ε_{rms}	ρ	ε_{avg}	ε_{rms}	ρ
I	1.30	1.57	0.99	0.72	0.91	0.99
P	2.07	2.52	0.97	0.82	1.09	0.98
B	2.79	3.22	0.97	0.87	1.12	0.98
All	2.50	3.96	0.97	0.84	1.10	0.98

as described throughout the paper. Additionally, the algorithm addresses the "all coefficients quantized to zero" problem by imposing a bound on the value of λ in those situations. Based on information provided by the author, these bounds have been set to the maximum value of λ found in lower frequencies, since it is not likely to get smaller values of λ as frequency increases.

Table II depicts a performance comparison of the proposed method with the implementation of [16]. The input for both methods is the full set of encoded video sequences. The symbols ε_{avg}, ε_{rms} and ρ represent, respectively, the average error, the root mean square error and the correlation, between true and estimated PSNR values. As can be observed from the table, the proposed method shows higher PSNR estimation accuracy regardless of the frame type.

D. Objective Quality assessment

The results for quality assessment have been evaluated by comparing the quality scores retrieved by the algorithm with the ones that result from the subjective tests.

Figure 9-a) depicts the the value of the propose perceptual distortion metric, D_g, that results from (35), versus the corresponding true MOS values. Following a procedure suggest by the *Video Quality Experts Group* (VQEG) in [44], a logistic function was used in order to map the D_g values into the MOS range 1–5, used in the experiments. The estimated MOS values are therefore the result of:

$$\text{Estimated MOS} = a_0 + \frac{a_1}{1 + e^{a_2 + a_3 D_g}}, \qquad (38)$$

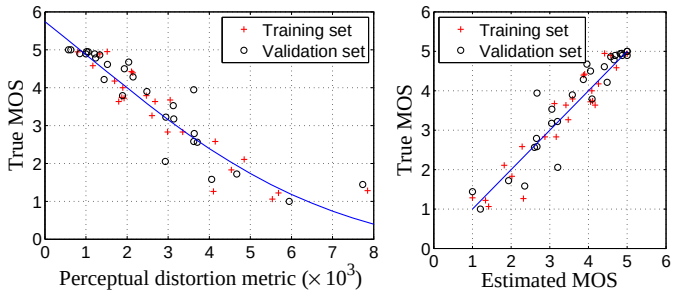
where a_0 to a_3 are curve fitting parameters. In order to compute these parameters, the available data points have been split into training and validation sets, using one half of the samples for each set. Parameter values are those that result from minimizing the square differences between true and estimated MOS scores in the training set, using the *Levenberg-Marquardt* method. A sketch of the resulting curve is also depicted in figure 9-a). Note that this procedure implicitly accounts for the influence of other perceptual factors, such as the effect of the deblocking filter used in the H.264 standard. Figure 9-b) shows the resulting normalized MOS estimates versus their true values. As can be observed, the NR objective quality scores resulting from the proposed algorithm are well correlated with the subjective quality assessment data.

In [44], VQEG suggests a set of statistical measurements in order to benchmark the performance of an objective metric. These performance indicators have been computed using the validation set and can be observed in table III. Pearson correlation and Spearman rank order coefficients are both above 0.9, which is a good result for video. The RMS is smaller than 0.4, which means that most of the MOS estimates computed by the metric are within the grades given by the observers.

Compared with other results found on the literature, the proposed method seems to outperform algorithms designed for similar purposes. In [13], *Ries et al.* propose a no-reference video quality assessment metric where the quality scores result from combining a set of motion features extracted at the decoder. The method is improved in [45], where a different parametrization for estimating MOS is used according to a previous classification of the video content. These methods were evaluated using SIF (352×240) H.264 encoded video sequences, and the declared performance in [13] and [45] are $CC = 0.80$ and $CC = 0.86$, respectively, which are below the results of the method proposed in this paper.

In [20], *Oelbaum* and *Diepold* propose a reduced reference method for H.264 encoded sequences where several features extracted from the video are combined (most of them are artifact measurements and motion oriented features), and the results are adjusted based on two parameter values sent through a side channel. The declared performance of this method is $CC = 0.84$, $RC = 0.80$ and $OR = 0.58$, which are also below the results achieved by the algorithm proposed in this paper.

A standard for reduced reference quality assessment of cable television signals is given in Recommendation ITU-T J.246 [21]. This metric – *Edge-PSNR* – is based on edge maps extracted from the original signals, which are sent to the receiver. The performance of this metric increases as the



(a) (b)

Fig. 9. MOS estimation results. (a) Perceptual distortion metric *vs*. true MOS values. (b) Estimated *vs*. true MOS values.

TABLE III
EVALUATION OF THE PROPOSED METRIC.

Root mean square error (RMS)	0.441
Pearson correlation coefficient (CC)	0.938
Spearman rank order coefficient (RC)	0.949
Outliers ratio (OR)	0.071

side channel bandwidth increases (*i.e.*, as the number of points in the sent edge map increases). The resulting values for CC are in the range $0.81 - 0.83$. Again, our method shows better performance. However, it must be kept in mind that the method proposed in this paper is adapted to DCT-based video encoding while the standardized method [21] is not distortion specific.

VI. CONCLUSIONS

A no-reference quality assessment algorithm for H.264/AVC encoded video sequences has been proposed. The algorithm comprises a local error estimation module followed by an error weighting module based on a perceptual spatio-temporal model.

The error estimation module is able to compute PSNR estimates based on the quantization steps and DCT coefficient values taken from an H.264 bit stream. The results of this module outperform the state-of-the-art algorithm in [16]. The no-reference quality scores are then computed based on the error estimates and on the motion vectors extracted from the bit stream. These MOS estimates correlate well with the human perception of quality and show better results than other algorithms, derived with the same purpose, found in literature.

As for future work, the algorithm should be extended in order to deal with transmission errors (*i.e.*, packet losses). Another topic that could increase the performance of the algorithm is the introduction of luminance and local contrast error masking, using a more complete HVS perceptual model.

ACKNOWLEDGMENT

The authors would like to thank Arnd Eden for providing them additional details for the implementation of his algorithm.

REFERENCES

[1] H. R. Wu and K. R. Rao, *Digital Video Image Quality and Perceptual Vision*. CRC Press, 2006.

[2] G. Ghinea, G.-M. Muntean, P. Frossard, M. Etoh, F. Speranza, and H. Wu, "IEEE Transactions on Broadcasting – Special issue on "quality issues on mobile multimedia broadcasting", vol. 7, no. 3, part II," September 2008.

[3] ITU-R, "Recommendation BT.500-11 – Methodology for the subjective assessment of the quality of television pictures," 1974–2002.

[4] ITU-T, "Recommendation P.910 – Subjective video quality assessment methods for multimedia applications," 1999.

[5] S. Winkler, "A perceptual distortion metric for digital color video," in *proc. of SPIE*, vol. 3644, S. Jose, USA, 1999, pp. 175–184.

[6] Z. Wang, L. Lu, and A. C. Bovik, "Video quality assessment based on structural distortion measurement," *Image Communication - Special issue on Objective Video Quality Metrics*, vol. 19, no. 2, pp. 121–132, February 2004.

[7] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, April 2004.

[8] E. Ong, X. Yang, W. Lin, Z. Lu, and S. Yao, "Perceptual quality metric for compressed videos," in *proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 2, Philadelphia, USA, March 2005, pp. 581–584.

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

[10] H. Wu and M. Yuen, "A generalized block-edge impairment metric for video coding," *IEEE Signal Processing Letters*, vol. 4, no. 11, pp. 317–320, November 1997.

[11] L. Meesters and J.-B. Martens, "A single-ended blockiness measure for JPEG-coded images," *Signal Processing*, vol. 82, no. 3, pp. 369–387, March 2002.

[12] F. Pan, X. Lin, S. Rahardja, W. Lin, E. Ong, S. Yao, Z. Lu, and X. Yang, "A locally adaptive algorithm for measuring blocking artifacts in images and videos," *Signal Processing: Image Communication*, vol. 19, no. 6, pp. 499–506, July 2004.

[13] M. Ries, O. Nemethova, and M. Rupp, "Motion based reference-free quality estimation for H.264/AVC video streaming," in *proc. of International Symposium on Wireless Pervasive Computing*, S. Juan, Puerto Rico, February 2007.

[14] D. S. Turaga, Y. Chen, and J. Caviedes, "No-reference PSNR estimation for compressed pictures," *Image Communication - Special issue on Objective Video Quality Metrics*, vol. 19, no. 2, pp. 173–184, February 2004.

[15] A. Ichigaya, M. Kurozumi, N. Hara, Y. Nishida, and E. Nakasu, "A method of estimating coding PSNR using quantized DCT coefficients," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 2, pp. 251–259, February 2006.

[16] A. Eden, "No-reference estimation of the coding PSNR for H.264-coded sequences," *IEEE Transactions on Consumer Electronics*, vol. 53, no. 2, pp. 667–674, May 2007.

[17] T. Brandão and M. P. Queluz, "No-reference PSNR estimation algorithm for H.264 encoded video sequences," in *proc. of EUSIPCO - European Signal Processing Conference*, Lausanne, Switzerland, August 2008.

[18] Z. Wang, G. Wu, H. Sheikh, E. Simoncelli, E.-H. Yang, and A. Bovik, "Quality-aware images," *IEEE Transactions on Image Processing*, vol. 15, no. 6, pp. 1680–1689, June 2006.

[19] M. Masry, S. Hemami, 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. 260–273, February 2006.

[20] T. Oelbaum and K. Diepold, "A reduced reference video quality metric for AVC/H.264," in *proc. of EUSIPCO - European Signal Processing Conference*, Poznan, Poland, September 2007, pp. 1265–1269.

[21] ITU-T, "Recommendation J.246 – Perceptual visual quality measurement techniques for multimedia services over digital cable television networks in the presence of a reduced bandwidth reference." 2008.

[22] B. Girod, "What's wrong with mean-square error?" in *Digital Images and Human Vision*, A. B. Watson, Ed. MIT Press, 1993.

[23] T. Brandão and M. P. Queluz, "No-reference image quality assessment based on DCT domain statistics," *Signal Processing*, vol. 88, no. 4, pp. 822–833, April 2008.

[24] A. B. Watson, "DCT quantization matrices optimized for individual images," in *proc. of SPIE Human Vision, Visual Processing, and Digital Display IV*, S. Jose, USA, 1993.

[25] ITU-T, "Recommendation H.264 – Advanced video coding for generic audiovisual services," 2005.

[26] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer, and T. Wedi, "Video coding with H.264/AVC: tools, performance, and complexity," *Circuits and Systems Magazine, IEEE*, vol. 4, no. 1, pp. 7–28, Quarter 2004.
[27] D. H. Kelly, "Motion and vision II: stabilized spatio-temporal threshold surface," *Journal of the Optical Society of America*, vol. 69, no. 10, pp. 1340–1349, October 1979.
[28] S. Daly, "Engineering observations from spatiovelocity and spatiotemporal visual models," in *Vision model and applications to image and video processing*, C. van den Branden Lambrecht, Ed. Kluwer, 2001.
[29] R. Duda, P. Hart, and D. Stork, *Pattern Classification - 2nd Edition*. Wiley-Interscience, 2000.
[30] Heinrich-Hertz-Institut, "JM 12.4 – H.264 reference software," December 2007, available online at http://iphome.hhi.de/suehring/tml/.
[31] E. Lam and J. Goodman, "A mathematical analysis of the DCT coefficient distributions for images," *IEEE Transactions on Image Processing*, vol. 9, no. 10, pp. 1661–1666, October 2000.
[32] J. Eggerton and M. Srinath, "Statistical distributions of image DCT coefficients," *Computers & Electrical Engineering*, vol. 12, no. 3–4, pp. 137–145, January 1986.
[33] Y. Altunbasak and N. Kamaci, "An analysis of the DCT coefficient distribution with the H.264 video coder," in *proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 3, Montreal, Canada, May 2004, pp. 177–180.
[34] F. Muller, "Distribution shape of two-dimensional DCT coefficients of natural images," *Electronic Letters*, vol. 29, no. 22, pp. 1935–1936, October 1993.
[35] T. Eude, R. Grisel, H. Cherifi, and R. Debrie, "On the distribution of the DCT coefficients," in *proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 5, Adelaide, Australia, April 1994, pp. 365–368.
[36] J.-H. Chang, J. W. Shin, N. S. Kim, and S. Mitra, "Image probability distribution based on generalized gamma function," *IEEE Signal Processing Letters*, vol. 12, no. 4, pp. 325–328, April 2005.
[37] J. Price and M. Rabbani, "Biased reconstruction for JPEG decoding," *IEEE Signal Processing Letters*, vol. 6, no. 12, pp. 297–299, December 1999.
[38] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. Springer, 2001.
[39] S. Winkler, *Digital video quality*. Wiley, 2005.
[40] J. G. Robson, "Spatial and temporal contrast sensitivity functions of the visual system," *Journal of the Optical Society of America*, vol. 56, pp. 1141–1142, 1966.
[41] J. Yang and W. Makous, "Spatio-temporal separability in contrast sensitivity," *Vision Research*, vol. 34, no. 19, pp. 2569–2576, 1994.
[42] R. Carpenter, *Movements of the eyes*. Pion, 1988.
[43] C. Lambrecht, "Perceptual model and architectures for video coding applications," Ph.D. dissertation, École Polytechnique Fédérale de Lausanne, 1996.
[44] VQEG, "Final report from the video quality experts group on the validation of objective models of video quality assessment, phase II," www.vqeg.org, Tech. Rep., August 2003.
[45] M. Ries, O. Nemethova, and M. Rupp, "Performance evaluation of mobile video quality estimators," in *proc. of EUSIPCO - European Signal Processing Conference*, Poznan, Poland, September 2007.

Tomás Brandão was born in Lisbon, Portugal, in 1975. He received the *Licenciatura* and M.S. degrees in Electrical and Computer Engineering from the Instituto Superior Técnico (IST), Technical University of Lisbon, Portugal, in 1999 and 2002, respectively. In 2006 he enrolled in the PhD program at the Technical University of Lisbon. He is currently a senior assistant lecturer at the ISCTE-IUL, where is has been lecturing Computer Architecture courses since 2002. He is also a researcher at the Instituto de Telecomunicações, Lisbon Portugal. His main research interests are digital signal and image processing.

Maria Paula Queluz received the B.S. and the M.S. degrees in Electrical and Computer Engineering from the Instituto Superior Técnico (IST), Technical University of Lisbon, Portugal, in 1985 and 1989 respectively, and the PhD degree from the Catholic University of Louvain, Louvain-la-Neuve, Belgium, in 1996. Since 1985, she has been with the Department of Electrical and Computer Engineering, IST, where she is currently Assistant Professor. She is also a research member at the Instituto de Telecomunicações, Lisbon, Portugal. Her main scientific interests include image analysis/processing, copyright protection and mobile communications.