

No-Reference Pixel Video Quality Monitoring of Channel-Induced Distortion

Giuseppe Valenzise *Student Member, IEEE*, Stefano Magni, Marco Tagliasacchi *Member, IEEE*, Stefano Tubaro *Member, IEEE*

Abstract— Video transmitted over an error-prone network may be received at the decoder with degradations due to packet losses. No-reference (NR) quality monitoring algorithms are the most practical way to measure the quality of the received video, since they do not impose any change with respect to the network architecture. Conventionally, these methods assume the availability of the corrupted bitstream. In some situations this is not possible, e.g. because the bitstream is encrypted or processed by third-party decoders, and only the decoded pixel values can be used. The major issue in this scenario is the lack of knowledge about which regions of the video have been actually lost, which is a fundamental ingredient for estimating channel-induced distortion. In this paper we propose a maximum-a-posteriori estimation of the pattern of lost macroblocks, which assumes the knowledge of the decoded pixels only. This information can be used as input to a no-reference quality monitoring system, which produces an accurate estimate of the MSE distortion introduced by channel errors. The results of the proposed method are well correlated with the MSE distortion computed in full-reference mode, with a linear correlation coefficient equal to 0.9 at frame level and 0.98 at sequence level.

Index Terms— Quality monitoring, No-Reference, channel errors

I. INTRODUCTION

When video is distributed over a packet network, the visual quality with respect to the original can be degraded by channel errors, delay or jitter suffered during transmission. However, in practical in-network video quality monitoring the original reference signal is typically not available for comparison at the decoder. In order to overcome this problem, reduced-reference (RR) and no-reference (NR) approaches have been proposed in the literature. The first ones postulate the availability of at least a compact representation of the original signal, which is sent over an auxiliary channel to the user, where it is matched against the received signal to produce an estimate of the distortion or a prediction of users' mean opinion score (MOS) [2][3][4]. Conversely, no-reference methods do not rely on the availability of the original video. Instead, they make

assumptions on the way video sequences are coded and/or transmitted. Thus, quality evaluation is highly challenging, but deployment does not require any change in the network architecture. For this reason, no-reference methods have been attracting a great deal of attention in the past few years.¹

No-reference quality monitoring techniques can be classified into methods that leverage only decoded-pixel information (no-reference pixel, NR-P) and approaches that extract information from the received bitstream, without performing full decoding (no-reference bitstream, NR-B). Also, there are systems that use both sources of information, which are known as “hybrid” and are denoted in the following as NR-BP. In some circumstances, in-network monitoring might not be possible, e.g. if the network is regulated by local-loop unbundling or shared access. In this case, the service provider has no control on the bitstream at intermediate nodes, since these belong to the owner of the network. Furthermore, the bitstream may be unavailable because it was encrypted, or the client employs a third-party hardware decoder, which may be very expensive to adapt to the quality monitoring task. In these cases, only the pixel values of the decoded video sequence can be used and, consequently, only the NR-P approach is feasible. With respect to NR-B, NR-P methods have the considerable advantage of having the decoded pixels available. However, the original bitstream parameters and the location of the errors are not known and must be somehow inferred from the decoded video. This can be a very difficult task, because video degradations can be confused with the original frame content, especially when sophisticated error concealment strategies are employed at the decoder.

In our previous work [1], we considered a NR-P method, where we addressed the problem of estimating which portions of a frame have been lost during transmission. Then, we used this information to compute the MSE distortion for H.264/AVC video. We showed that lost macroblocks can be identified, to some extent, by looking at footprints left by the error concealment process. Specifically, we concentrated on temporal error concealment only, and deemed as lost all those blocks that had an identical predictor in a reference frame. In order to reduce the impact of false positives, e.g. skipped macroblocks incorrectly labeled as lost, we considered in [1] a morphological filter, which assumed the prior knowledge of the exact slicing structure.

¹In this paper, we focus on no-reference methods aimed at estimating the distortion introduced by channel errors. We refer the reader to Section II for a brief discussion on no-reference methods that estimate artifacts introduced by lossy coding.

In this paper we generalize and extend our previous work. Similarly to [1], we consider a NR-P setting, which implies the availability of the decoded (and error-concealed) video pixels only. At the same time, we relax the constraints on the necessary prior knowledge: i) we do not assume to know the specifications of the error concealment algorithm used, but only the broad category it belongs to (handling both spatial and temporal concealment); ii) we do not assume to know the exact slicing structure, but only that slices follow some regular pattern (raster, interleaved, wipe, checkerboard, etc.) and are not composed by a single macroblock. To the authors' knowledge, there are no other NR-P methods for estimating channel-induced distortion in the literature which do not make specific assumptions on the adopted slicing structure. Hence, as first contribution of this paper, we formulate a flexible *maximum a posteriori* (MAP) estimation of the location of corrupted macroblocks. Throughout the paper, we will use the example of H.264/AVC video, as this coding standard has demonstrated superior coding performance in comparison to previous codecs [5], and it is especially suitable for transmitting video over IP [6] and wireless [7] networks. Nevertheless, we remark that the applicability of the proposed method MAP estimation technique is not restricted to a specific video coding standard, but can be employed in any block-based hybrid video codec.

Having an accurate estimate of the support of channel losses is extremely relevant in NR quality assessment. In fact, several NR methods (see Section II) assume the availability of this information to produce accurate estimates of the distortion. As second contribution of this paper, we show how a NR-BP distortion estimation algorithm designed for H.264/AVC video can be converted into a NR-P method, using the proposed MAP estimation of channel losses to replace the lack of bitstream information. Our experiments show that channel-induced distortion can be estimated in a NR-P fashion without significant loss of accuracy with respect to the NR-BP scenario.

The rest of this paper is organized as follows. Section II reviews previous work in the literature related to NR quality monitoring, focusing on distortion introduced by channel errors. Section III describes in detail the problem of estimating the portions of a video frame that have been lost. In Section IV, we derive and evaluate the accuracy of the maximum-a-posteriori estimator of the location of corrupted macroblocks. This estimate is used in Section V to enable a NR-BP quality monitoring system to work effectively in the absence of the bitstream. Finally, Section VI concludes the paper.

II. RELATED WORK

A number of no-reference metrics have been proposed for quantifying blurring or blocking artifacts in compressed images (e.g. [8], [9], [10]). For example, the NR-P method in [8] evaluates the distortion introduced by video coding by automatically and perceptually quantifying blocking artifacts of the Discrete Cosine Transform (DCT) coded macroblocks. The NR-P metric in [9] estimates blurring in an image based on the average width of its edges. Many of these methods can be directly adapted to the case of video [11], even though

other impairments due to motion (e.g. motion-compensated edge artifacts [12]) may occur that are not present in still images. Ichigaya et al. [13], [14] and Brandão and Queluz [15], [16] propose NR-BP methods to estimate the PSNR of an MPEG2 or H.264/AVC-coded video sequence based on the distribution of its quantized DCT coefficients. To this end, they assume the availability of some bitstream information such as the quantization parameter (QP) adopted to quantize the signal.

For the case of packet-loss impairments (PLI), the absence of bitstream information may be a major limitation, as mentioned above. Thus, most of the research in the past decade has concentrated on NR-B and NR-BP metrics (see e.g. [17], [18], [19], [20], [21], [22] and references therein). We can basically identify three classes of NR techniques for PLI, according to the type of result they return.

A. Estimating perceptual quality.

In the first category fall methods that aim at estimating the perceptual quality of a video sequence [23], [24], [25], [18], [22] from a set of features that can be extracted from the bitstream and/or the decoded pixels. The authors of [24] assess the fluidity impairments due to packet losses/jitter, bitrate adaption or cell losses in wireless cellular transmission, achieving linear correlations with MOS higher than 0.9. The work in [25] quantifies PLI by weighting three factors: the PSNR drop due to the loss (error severity), the error length (duration of the loss including error propagation), and the “forgiveness effect”, i.e. viewers tend to forget impairments when, immediately later, the video is uncorrupted for a sufficient amount of time. The resulting metric is full-reference in nature, but can be easily extended to a NR setting by estimating the initial PSNR drop blindly (see methods below). The technique described in this paper would be beneficial for that purpose. The work in [26] estimates the overall quality of a transmitted video by considering the joint contribution of: picture distortion due to quantization; channel-induced distortion; and temporal masking effects of the human visual system. The importance of temporal variations in the perceived quality of a video has been analyzed in [27], where it is shown that including temporal variation metrics into standard PSNR or SSIM metrics would increase their accuracy for the case of packet loss distortion. The authors of [28] measure the MOS drop due to rebuffering interruptions and packet losses in QCIF content transmitted over a mobile network. A NR-P technique to evaluate PLI is described in [29]; however, this metric assumes the knowledge of slicing structure, which might be unknown in practice, and the results seem satisfactory with very simple error-concealment methods only.

B. Estimating MSE distortion.

The output of perceptual quality metrics is typically correlated to MOS values collected in subjective evaluation campaigns, in order to assess their accuracy. However, MOS values are meaningful only at the sequence level, i.e., over periods of time of at least a few seconds. In some applications, it may be useful to have an indication of the actual video fidelity at a finer granularity (e.g. over a single frame or at

the macroblock level). Instead of directly targeting estimation of visual quality, the approaches in this category focus on predicting the MSE distortion (and derived metrics such as PSNR) due to the channel losses [17], [30], [21]. MSE can be computed at any granularity from pixel to sequence. Therefore, it is flexible enough to be used for purposes different from just giving an overall score of the corrupted video. Nevertheless, PSNR can still be a good predictor of subjective video quality (in terms of MOS), in the case of PLI [31]. The NR-B method in [17] parses the bitstream at different levels of depth (full, quick or no-parse) in order to estimate the MSE distortion. Since there is no need for full decoding, this method is particularly suitable from the network provider standpoint for in-network quality monitoring. The NR-BP methods in [30] and [21], in contrast, are designed to work at the decoder side with little computational overhead. Yamada et al. [30] observe that the error concealment algorithm may perform more or less effectively depending on several factors, e.g., motion complexity and local texturing of the lost macroblock. Therefore, they propose a NR-BP heuristic that measures the *error concealment effectiveness*. When using zero-motion error concealment, the authors show accurate correlations with ground-truth MSE at the sequence level. This work has been recently extended in [18] to directly predict MOS using the smallest amount of information that can be deduced from an encrypted bitstream, e.g. the size of the payload. The resulting correlations at the sequence level, with subjective MOS scores gathered on high-resolution video, are higher than 0.85. Naccari et al. [21] describe a NO-Reference quality Monitoring (NORM) system which takes as input a H.264/AVC compliant bitstream together with the decoded (and error-concealed) frames to estimate the MSE distortion at the macroblock level. The reported correlations with the actual MSE are always above 0.8, at the macroblock (MB) level, and higher than 0.9 at the sequence level.

C. Estimating PLI visibility.

Finally, the third class of methods follow a different approach and aim at predicting the visibility of *each individual packet loss* [32], [19], [20], [33]. The task in this case is not to detect possible losses and predict their effect on quality, but rather to assess whether a loss (which is known to have occurred) is likely to be seen by a certain fraction of viewers. In [32] machine learning classifiers are used to predict packet loss visibility in MPEG-2 coded bitstreams. Similar techniques have been adapted to H.264/AVC coded bitstreams in [34], and extended to include further descriptors such as the Slice Boundary Mismatch (SBM), in order to quantify the spatial impairment introduced by error-concealment [19]. An interesting follow-up of this work [33] considers also the effect of scene cuts and camera motion on the visibility of losses.

III. PROBLEM OVERVIEW

Let us consider a video sequence, coded with a hybrid, block-based codec such as MPEG-2 or H.264/AVC. The sequence is then transmitted over an error-prone channel. Our goal is to estimate which portions of the video sequence have

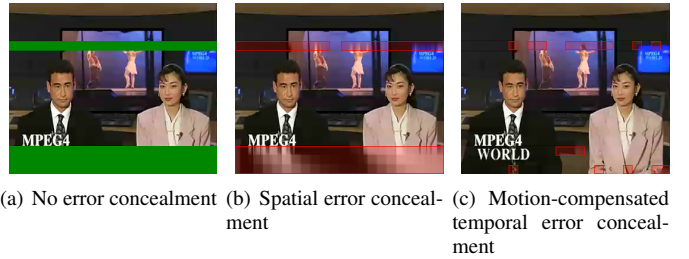


Fig. 1. The support of the initial error does not always coincide with the actual lost macroblocks (shown in (a)). Depending on the adopted error concealment strategy, the actual macroblocks with $d^I \neq 0$ (highlighted by the red-shaded boxes in the figure) may be substantially fewer.

been lost during transmission because of channel errors, given the decoded pixels only. We adopt the following notation. Let X denote the original video signal and b its encoded bitstream that is transmitted over the error-prone network. Let \hat{X} be the decoded video when b is received correctly, i.e. the video sequence reconstructed at the encoder. Similarly, let \tilde{b} denote the bitstream actually received at the decoder, which may be incomplete due to packet losses, and \tilde{X} the decoded video reconstructed after error concealment. When a macroblock i is lost, it is partially recovered by means of an error concealment algorithm. Perfect recovery of \hat{X} is in general not possible, resulting in a channel-induced distortion

$$d_i(t) = \frac{1}{B^2} \sum_{j=1}^B \sum_{k=1}^B \left(\hat{X}_i(j, k, t) - \tilde{X}_i(j, k, t) \right)^2, \quad (1)$$

where B is the macroblock size, $t = 1, \dots, T$ is a frame index, $\tilde{X}_i(j, k, t)$ is the pixel in position (j, k) of macroblock i ($i = 1, \dots, N$) in the decoded sequence, N is the total number of macroblocks in a frame, and $\hat{X}_i(j, k, t)$ is a pixel of macroblock i in the error-free video. Hereafter, estimated quantities will be denoted by placing a caret over the corresponding symbol, e.g. \hat{d} , in order to distinguish them from the true, unknown parameters.

We can decompose the distortion $d_i(t)$ due to a packet loss for a given macroblock i at time t into two components: an *innovation* and a *propagation* term,

$$d_i(t) = d_i^I(t) + d_i^P(t_1, \dots, t_n), \quad (2)$$

where t_l is the index of the l -th reference frame. In (2), $d_i^I(t)$ is the distortion produced by the lack of the original predictor and prediction residuals at time t and by the adopted error concealment algorithm. This term is often referred to as *initial error*, and can be measured, e.g., in terms of MSE or SSIM [19]. The term $d_i^P(t)$ accounts for the propagation of errors from the previous n reference frames due to the predictive nature of video codecs [35]. The model in (2) is flexible enough to take into consideration I, P or B slices: in the case of I frames, the term $d_i^P(t)$ in (2) accounts for spatial prediction in the same frame only, whereas for the case of B slices it will depend in general from both past and future frames, according to the adopted GOP structure. Based on (2), the MSE distortion

pooled over a whole frame is:

$$d(t) = \sum_{\substack{\{j: \text{MB } j \text{ has} \\ \text{been lost}\}}} d_j^I(t) + \sum_{i=1}^N d_i^P(t_1, \dots, t_n). \quad (3)$$

Notice that only the MBs lost at time t contribute to the initial error, though all the macroblocks of the frame can be indirectly affected by PLI due to error propagation from the reference frames. We can rewrite (3) in a more convenient form by introducing an indicator function $\mathcal{I}(t)$ of the losses at time t . Thus, (3) becomes:

$$d(t) = \sum_{i=1}^N [\mathcal{I}_i(t) d_i^I(t) + d_i^P(t_1, \dots, t_n)], \quad (4)$$

where the vector $\mathcal{I}(t) \in \{0, 1\}^N$ is defined as:

$$\mathcal{I}_i(t) = \begin{cases} 1 & \text{if MB } i \text{ has been lost} \\ 0 & \text{if MB } i \text{ has been correctly received.} \end{cases} \quad (5)$$

In the following, we will refer to $\mathcal{I}(t)$ as the pattern of lost macroblocks.

It turns out that identifying *exactly* which slices have been lost in a frame is an ill-posed problem, in the sense that it is not possible from decoded pixels only to reconstruct the original pattern of the loss $\mathcal{I}(t)$. The example in Figure 1 illustrates this phenomenon, showing how the ill-posed nature of the problem is related to the adopted error concealment algorithm. Notice that lost macroblocks which are perfectly restored by the error concealment algorithm do not leave any trace in the decoded pixels. In general, perfectly healed macroblocks are the ones whose content can be predicted exactly from their spatial/temporal neighbors (depending on the type of error concealment used). This is often the case when the original sequence has high spatial/temporal correlation and is coded at low bit rates. Also, more sophisticated error concealment algorithms are more likely to perfectly recover the lost macroblock content. As a result, we need to confine our attention to identifying only those macroblocks i which have been lost *and* have not been recovered perfectly by the error concealment algorithm. Let $\mathcal{D}(t)$ be a binary vector, whose elements $\mathcal{D}_i(t)$ are equal to one if $d_i^I(t) > 0$, and equal to zero otherwise. Therefore, we denote the binary vector of lost macroblocks producing non-zero initial distortion as $\mathcal{S}(t)$:

$$\mathcal{S}_i(t) = \begin{cases} 1 & \text{if } \mathcal{I}_i(t) = 1 \text{ and } \mathcal{D}_i(t) = 1 \\ 0 & \text{otherwise,} \end{cases} \quad (6)$$

and refer to it as the *support of the initial error*, to emphasize that these are the only MBs that contribute to channel-induced distortion. Indeed, these are the only macroblocks that are relevant from a quality monitoring standpoint.

IV. MAP ESTIMATION OF THE SUPPORT OF THE INITIAL ERROR

In this section we propose a flexible Bayesian approach to estimate \mathcal{S} , which incorporates both features observed from the decoded pixels and an *a priori* distribution $p(\mathcal{S})$ over all possible error patterns. This prior distribution may express,

for instance, very loose assumptions over the adopted slicing structure, e.g. that a slice consists of an unspecified number of horizontally contiguous macroblocks.

A. Definition of the posterior distribution for \mathcal{S}

Let $\mathbf{x}_i(t)$, $i = 1, \dots, N$, be a vector of features computed for each macroblock i of the observed frame $\tilde{X}(t)$. These features, which are further discussed in Section IV-B, may quantify e.g. the spatial complexity of the current macroblock, or the temporal activity in a neighborhood of the macroblock, etc. We collect the feature vectors of each MB in frame t into a single vector $\mathbf{x}(t) = [\mathbf{x}_1(t), \dots, \mathbf{x}_N(t)]$; likewise, we denote with $\mathbf{x} = [\mathbf{x}(1), \dots, \mathbf{x}(T)]$ the global feature vector for the entire video sequence. In the following, we assume that quantities without a temporal index t (e.g., \mathcal{S}) refer to the whole set of frames. The *conditional likelihood* $p(\mathbf{x}|\mathcal{S})$ expresses the probability of observing a feature vector \mathbf{x} conditioned on the (hidden) true pattern of errors \mathcal{S} (notice that this formulation embeds also the possible temporal dependence between \mathbf{x} and \mathcal{S}). We model \mathcal{S} as a binary-valued vector random variable with prior distribution $p(\mathcal{S})$. Using Bayes' rule, we can write the posterior distribution of \mathcal{S} given the feature vector \mathbf{x} as:

$$p(\mathcal{S}|\mathbf{x}) = \frac{p(\mathbf{x}|\mathcal{S})p(\mathcal{S})}{p(\mathbf{x})} \quad (7)$$

The MAP estimate of \mathcal{S} is that pattern of lost macroblocks $\hat{\mathcal{S}}$ which maximizes $p(\mathcal{S}|\mathbf{x})$. That is

$$\hat{\mathcal{S}} = \underset{\mathcal{S} \in \{0,1\}^{NT}}{\operatorname{argmax}} p(\mathcal{S}|\mathbf{x}) = \underset{\mathcal{S} \in \{0,1\}^{NT}}{\operatorname{argmax}} p(\mathbf{x}|\mathcal{S})p(\mathcal{S}) \quad (8)$$

where the maximization is over the set of all 2^{NT} possible patterns. Note that the factor $p(\mathbf{x})$ does not influence the MAP decision, thus it needs not be estimated.

In (7), the conditional dependence between \mathbf{x} and \mathcal{S} is not limited to a single frame, but it may encompass a larger temporal horizon. This setting captures the temporal dependencies in channel errors (e.g., losses in packet networks typically come in bursts whose length might be of several frames [36]). However, in order to simplify the model of the distribution in (7), we assume statistical independence from frame to frame, i.e.:

$$p(\mathcal{S}|\mathbf{x}) = \prod_{t=1}^T p(\mathcal{S}(t)|\mathbf{x}(t)) \quad (9)$$

In this way, the maximization of $p(\mathcal{S}|\mathbf{x})$ reduces to the separate optimization of each term $p(\mathcal{S}(t)|\mathbf{x}(t))$. We also make the following two assumptions. First, feature vectors $\mathbf{x}_1(t), \dots, \mathbf{x}_N(t)$ are conditionally independent given $\mathcal{S}(t)$, and each vector has a conditional density function $p(\mathbf{x}_i(t)|\mathcal{S}_i(t))$, which depends on $\mathcal{S}(t)$ only through $\mathcal{S}_i(t)$. Then, the likelihood function may be rewritten as:

$$p(\mathbf{x}(t)|\mathcal{S}(t)) = \prod_{i=1}^N p(\mathbf{x}_i(t)|\mathcal{S}_i(t)). \quad (10)$$

Second, we observe that a single packet drop entails the loss of a whole slice, which is rarely composed by a single macroblock, but of 4- or 8-connected groups of macroblocks.

These schemes include raster scan, interleaved, wipe, checkerboard and other widely employed slicing patterns. In other terms, it is reasonable to penalize those configurations $\mathcal{S}(t)$ having isolated macroblocks marked as lost. On this basis, we model the prior distribution $p(\mathcal{S}(t))$ as a pairwise interaction Markov Random Field (MRF) [37] of the form:

$$p(\mathcal{S}(t)) \propto \exp \left\{ \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N w_{ij}(t) \delta(\mathcal{S}_i(t), \mathcal{S}_j(t)) \right\}, \quad (11)$$

where: $w_{ii} = 0$ and $w_{ij} = w_{ji} \geq 0$, with the strict inequality holding only if i and j are neighboring macroblocks (in the 4- or 8-connectivity sense); and $\delta(\mathcal{S}_i(t), \mathcal{S}_j(t))$ is the Kronecker delta function which is equal to one if $\mathcal{S}_i(t) = \mathcal{S}_j(t)$ and is zero otherwise. The term inside the sum in (11), which is also known as *interaction potential* [38], is different from zero only if i and j are neighbors and $\mathcal{S}_i(t) = \mathcal{S}_j(t)$; in this manner, we favor those configurations where neighboring macroblocks are either all lost or all received. Notice that the formulation in (11) can be easily extended to embed further prior knowledge; e.g., if the probability of losing a packet is known, it would be straightforward to include this knowledge, e.g., multiplying (11) by the packet loss rate (PLR).

Apart from an additive constant, the log-posterior distribution $\ln p(\mathcal{S}(t)|\mathbf{x}(t))$ can be written as [39]:

$$\begin{aligned} \ln p(\mathcal{S}(t)|\mathbf{x}(t)) = & \sum_{i=1}^N \lambda_i(t) \mathcal{S}_i(t) \\ & + \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N w_{ij}(t) \delta(\mathcal{S}_i(t), \mathcal{S}_j(t)), \quad (12) \end{aligned}$$

where $\lambda_i(t) = \ln \left[\frac{p(\mathbf{x}_i|\mathcal{S}_i(t)=1)}{p(\mathbf{x}_i|\mathcal{S}_i(t)=0)} \right]$ is the log-likelihood ratio for macroblock i .

We deal with the efficient maximization of (12) in Section IV-D. Before that, we describe the features \mathbf{x} used to characterize impairments due to channel losses, and provide a functional form for their likelihood, for two kinds of error concealment algorithms.

B. Features and likelihood function

In order to estimate \mathcal{S} accurately, it is fundamental to select a relevant set of features \mathbf{x} , which capture the presence of badly concealed errors in the decoded video. We start observing that $\mathcal{S}_i(t)$ depends on: i) whether a MB has been lost; and ii) how well the error concealment algorithm recovered the lost content. Accordingly, we devise two classes of features, each targeting a specific set of macroblocks as illustrated in Figure 2.

The first kind of features aims at identifying which MBs might have been lost, i.e., MBs i such that $\mathcal{I}_i(t) = 1$. When a macroblock is lost and its content is reconstructed through an error concealment process, some characteristic traces are left in the decoded pixels, as detailed in Sections IV-B.1 and IV-B.2. For instance, the lost MB content is typically predicted from spatially or temporally neighboring MBs. As a result, the rendered pixels show characteristic (and predictable)

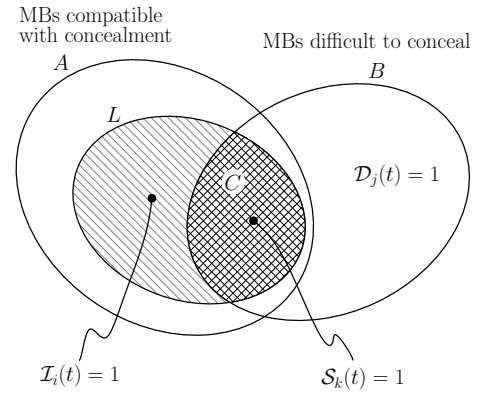


Fig. 2. The support of the initial error, represented by the set C , is included in a set obtained by intersecting the set A of macroblocks which are compatible with concealment, and the set B of macroblocks which, if lost, produce a positive initial distortion.

correlations with the neighboring macroblocks, revealing that they could be the result of an error concealment process. In the following, we denote by $\mathbf{x}_i^A(t)$ those features which describe such error concealment traces. Notice that the set of macroblocks *compatible* with error concealment may include as well many correctly received MBs, whose content is almost perfectly predictable from their neighbors. For this reason, the $\mathbf{x}_i^A(t)$ features tend to overestimate the number of lost macroblocks.

As extensively discussed above, lost macroblocks could be perfectly recovered (with zero distortion), provided that they were perfectly predictable from their neighbors. The second kind of features we propose aims at identifying those portions of a frame where errors are unlikely to be effectively concealed, due to the intrinsic complexity of the scene content. In other terms, we look for MBs i such that $\mathcal{D}_i(t) = 1$ (set B in Figure 2). Features that describe the concealment effectiveness are denoted as $\mathbf{x}_i^B(t)$.

The set of macroblocks i such that $\mathcal{S}_i(t) = 1$ is, approximately, the intersection of the sets of MBs described by the two categories of features introduced above, as depicted in Figure 2. Since features $\mathbf{x}_i^A(t)$ describe a property of the channel errors and of the error concealment, while $\mathbf{x}_i^B(t)$ describe a property of the *original* frame content, it is reasonable to consider them as statistically independent. That is,

$$\begin{aligned} p(\mathbf{x}_i(t)|\mathcal{S}_i(t)) &= p(\mathbf{x}_i^A(t)|\mathcal{S}_i(t)) \cdot p(\mathbf{x}_i^B(t)|\mathcal{S}_i(t)) \\ &= p(\mathbf{x}_i^A(t)|\mathcal{I}_i(t), \mathcal{D}_i(t)) \cdot p(\mathbf{x}_i^B(t)|\mathcal{I}_i(t), \mathcal{D}_i(t)) \\ &= p(\mathbf{x}_i^A(t)|\mathcal{I}_i(t)) \cdot p(\mathbf{x}_i^B(t)|\mathcal{D}_i(t)), \quad (13) \end{aligned}$$

where the simplification in the last equation comes from the fact that features $\mathbf{x}_i^A(t)$ are conditionally dependent on $\mathcal{I}_i(t)$ only, while $\mathbf{x}_i^B(t)$ depend solely on $\mathcal{D}_i(t)$.

For the sake of clarity, Figure 3 illustrates how different subsets of macroblocks are selected to approximate $\mathcal{S}(t)$, when spatial error concealment is considered. The true pattern of losses $\mathcal{I}(t)$ is represented by the highlighted area in Figure 3(a). For each macroblock of the decoded frame, we can compute the likelihood $p(\mathbf{x}_i^A(t)|\mathcal{S}_i(t) = 1) = p(\mathbf{x}_i^A(t)|\mathcal{I}_i(t) = 1)$; this is shown in Figure 3(b). As expected, all the lost

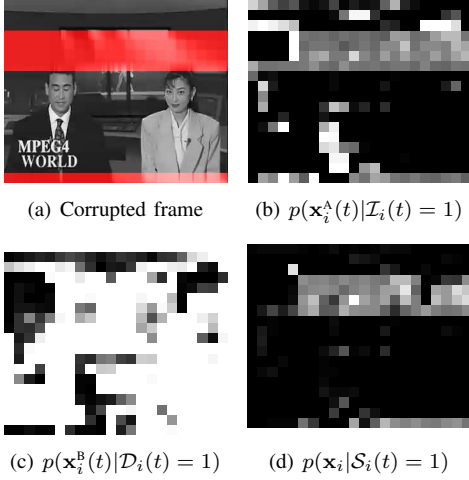


Fig. 3. Estimated likelihood for spatial concealment in the *News* sequence. Brighter macroblocks correspond to higher values of likelihood.

macroblocks have a large likelihood; however, there are several correctly received macroblocks which also exhibit large likelihood values. Nevertheless, in many of these blocks the errors would be easily concealed, so they should be filtered out. These blocks correspond to the set B and are highlighted in Figure 3(c). The intersection of the sets A and B corresponds to the macroblocks having a high joint probability $p(\mathbf{x}_i(t)|\mathcal{S}_i(t) = 1)$. These likelihood values for each macroblock are shown in Figure 3(d). We qualitatively observe that, in this case, the joint use of $\mathbf{x}_i^A(t)$ and $\mathbf{x}_i^B(t)$ can improve the accuracy of the detection, contributing to the reduction of false positives. In the following, we describe two instances of $\mathbf{x}_i^A(t)$ and $\mathbf{x}_i^B(t)$, which correspond to two widely used error concealment approaches: motion-compensated temporal error concealment and spatial error concealment.

1) *Motion-compensated temporal concealment*: When the decoder adopts motion-compensated temporal concealment (MCTC) for P or B frames, each lost macroblock is replaced by another macroblock in a previous reference frame. Specifically, in the MCTC strategy [40] proposed for H.264/AVC video, the concealment algorithm finds the macroblock in the reference frame that minimizes the side match distortion, i.e., the border discontinuity between the error-concealed macroblock and its neighbors. This procedure is repeated for each macroblock in the lost slice, starting from the slice boundaries (which can leverage correctly received and decoded pixels) towards the interior of the slice. The widely used zero-motion copy error concealment is a special case of MCTC.

We describe the footprints left by the error concealment for each MB i at frame t using the temporal prediction residual energy. First, we estimate motion vectors from the decoded video in order to find a motion-compensated predictor \tilde{X}_i^{MC} for each decoded macroblock \tilde{X}_i . Then,

$$\mathbf{x}_{i,\tau}^A(t) = \frac{1}{B^2} \sum_{j=1}^B \sum_{k=1}^B \left(\tilde{X}_i(j, k, t) - \tilde{X}_i^{\text{MC}}(j, k, t-1) \right)^2, \quad (14)$$

where the subscript τ in $\mathbf{x}_{i,\tau}^A(t)$ indicates the fact that we are considering temporal concealment. By construction,

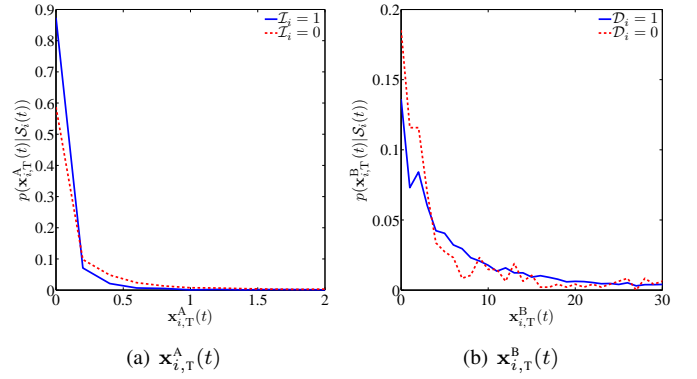


Fig. 4. Histograms (normalized) of the features related to temporal concealment $\mathbf{x}_{i,\tau}^A(t)$ and $\mathbf{x}_{i,\tau}^B(t)$ for the *Foreman* sequence.

macroblocks compatible with error concealment must have $\mathbf{x}_{i,\tau}^A(t) = 0$. In practice, due to the action of in-loop filtering (such as the deblocking filter in H.264/AVC), it may happen that the observed block does not match exactly its predictor. As an example, Figure 4(a) shows the conditional likelihood functions $p(\mathbf{x}_{i,\tau}^A(t)|\mathcal{S}_i(t) = 1)$ and $p(\mathbf{x}_{i,\tau}^A(t)|\mathcal{S}_i(t) = 0)$, for the *Foreman* sequence. The two functions are modeled by exponentials with different decay parameters:

$$\begin{aligned} p(\mathbf{x}_{i,\tau}^A(t)|\mathcal{I}_i(t) = 1) &\propto e^{-\alpha_{1,\tau}\mathbf{x}_{i,\tau}^A(t)} \\ p(\mathbf{x}_{i,\tau}^A(t)|\mathcal{I}_i(t) = 0) &\propto e^{-\alpha_{0,\tau}\mathbf{x}_{i,\tau}^A(t)} \end{aligned} \quad (15)$$

where $\alpha_{1,\tau} > \alpha_{0,\tau} > 0$.

In order to characterize the effectiveness of error concealment, we observe that, for MCTC, the complexity of the motion field around a lost macroblock plays a major role in determining the quality of the reconstructed signal. For a macroblock i , let $\{\overline{\text{MV}}_j\}$ be the set of motion vectors (estimated on the decoded video) of MBs j that are neighbors of MB i . We define $\mathbf{x}_{i,\tau}^B(t) = \text{var}\{\overline{\text{MV}}_j\}$ as the variance of the local motion field. MCTC is likely to be effective when $\mathbf{x}_{i,\tau}^B(t)$ is small. We show an example of the conditional distribution of $\mathbf{x}_{i,\tau}^B(t)$ in Figure 4(b). To draw this picture, we first corrupted a sequence. Then we computed the value of $\mathbf{x}_{i,\tau}^B(t)$ for those lost macroblocks which have a positive (or zero) channel-induced distortion. The two likelihood functions can be modeled using two exponentials with different decay parameters:

$$\begin{aligned} p(\mathbf{x}_{i,\tau}^B(t)|\mathcal{D}_i(t) = 1) &\propto e^{-\beta_{1,\tau}\mathbf{x}_{i,\tau}^B(t)} \\ p(\mathbf{x}_{i,\tau}^B(t)|\mathcal{D}_i(t) = 0) &\propto e^{-\beta_{0,\tau}\mathbf{x}_{i,\tau}^B(t)} \end{aligned} \quad (16)$$

where $\beta_{0,\tau} > \beta_{1,\tau} > 0$. In order to compute $\mathbf{x}_{i,\tau}^B(t)$, we need the original motion field, which is in general not available at the decoder. Therefore, we approximate it with the motion field of the previous frame. This approximation is valid if the motion field varies slowly along time. If the motion field in the previous frame does not approximate sufficiently well the current one, $\mathbf{x}_{i,\tau}^B(t)$ could actually be misleading and deteriorate the results, as explained in Section IV-C.

2) *Spatial concealment*: A common technique for spatial concealment (SC) predicts the missing content of a lost MB by interpolating the pixel values from pixels along the borders of neighboring MBs. Specifically, in the method proposed in

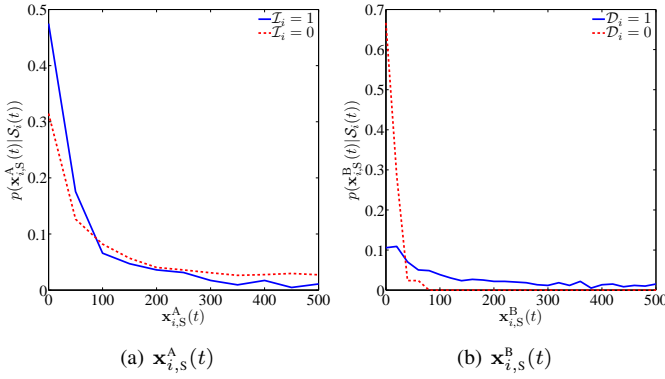


Fig. 5. Histograms (normalized) of the features related to spatial concealment $\mathbf{x}_{i,T}^A(t)$ and $\mathbf{x}_{i,T}^B(t)$ for the *Foreman* sequence.

[40] for H.264/AVC video, the missing pixels of a lost MB are replaced by bilinear interpolation using the border pixels of the 4 neighboring MBs. A larger weight is given to pixels close to the borders. We assume that the class of the interpolation kernel is known (e.g. bilinear, bicubic, etc.). Let \tilde{X}_i be the spatial predictor of a MB \tilde{X}_i . We define $\mathbf{x}_{i,s}^A(t)$ as the spatial prediction residual energy. That is:

$$\mathbf{x}_{i,s}^A(t) = \frac{1}{B^2} \sum_{j=1}^B \sum_{k=1}^B \left(\tilde{X}_i(j, k, t) - \tilde{X}_i^{\text{SP}}(j, k, t) \right)^2, \quad (17)$$

where the subscript s indicates that this feature refers to spatial concealment. Therefore, lost macroblocks in a spatially concealed frame, which are completely predictable from their neighbors, must have $\mathbf{x}_{i,s}^A(t) = 0$. The effect of in-loop filtering (e.g. deblocking) may actually change the content of the predicted block, thus $\mathbf{x}_{i,s}^A(t) \approx 0$. An example of the conditional distribution of $\mathbf{x}_{i,s}^A(t)$ is given in Figure 5(a). We model the conditional likelihood $p(\mathbf{x}_{i,s}^A(t)|\mathcal{S}_i)$ with two exponential distributions as in (15), with decay parameters $\alpha_{1,s} > \alpha_{0,s} > 0$.

As for the SC effectiveness, we observe that errors in those regions that are easily spatially predictable are likely to be concealed well. In order to identify which macroblocks are predicted effectively with spatial concealment, we should know the original frame content. Since this knowledge is not available, we approximate the uncorrupted frame at time t , $\tilde{X}(t)$, with the previously decoded frame $\tilde{X}(t-1)$. This approximation is valid when the spatially concealed frame does not correspond to a scene change, a sudden occlusion and there are no fast moving objects. Therefore, the concealment effectiveness feature $\mathbf{x}_{i,s}^B(t)$ is the spatial residual energy of the co-located MB in the previous frame. That is:

$$\mathbf{x}_{i,s}^B(t) = \frac{1}{B^2} \sum_{j=1}^B \sum_{k=1}^B \left(\tilde{X}_i(j, k, t-1) - \tilde{X}_i^{\text{SP}}(j, k, t-1) \right)^2. \quad (18)$$

An example of the conditional distribution of $\mathbf{x}_{i,s}^B(t)$ is depicted in Figure 5(b). The two likelihood functions can be modeled using two exponentials as in (16), with decay parameters $\beta_{0,s} > \beta_{1,s} > 0$.

TABLE I
CODING CONDITIONS ADOPTED IN THE EXPERIMENTS

Parameter	Value
Encoder	H.264/AVC reference software, main profile (JM 12.3)
Number of frames	300
Intra period	15 frames
Number of reference frames	5
QP I/P slices	Fixed, see Table II
Macroblock partitions for motion estimation	Enabled
Motion estimation algorithm	Enhanced predictive zonal search (EPZS)
Slicing mode	Fixed number of MBs
Slice group	Raster scan (1 slice = 1 row of macroblocks)

C. Discriminability of the features

We measure the discriminative power of a feature using the receiving operating characteristic (ROC) curve. Such plot is obtained by thresholding the value of the feature, in such a way that MBs where the value of the feature is larger (smaller) of the threshold are labeled as positive (negative). In such a way, we obtain a binary classification (*labeling*) of lost and received macroblocks. Sweeping several values of the threshold, it is possible to achieve different tradeoffs between true and false positive rates (TPR and FPR, respectively). TPR and FPR are defined as follows. The label *positive* is used to denote a macroblock which has been lost entailing a non-zero initial error. Conversely, a macroblock which has been received correctly, or which has been lost but has been perfectly restored by error concealment (i.e., with zero initial distortion), is labelled as *negative*. A *true positive* (TP) is a positive macroblock, which has been labeled as positive. A *false positive* (FP), instead, is a negative MB which has been labeled as positive. In a similar way, true negatives (TN) and false negatives (FN) can be defined. It is customary to normalize the absolute number of true or false positives to their rate, defined as:

$$TPR = TP/P = TP/(TP + FN) \quad (19)$$

$$FPR = FP/N = FP/(FP + TN), \quad (20)$$

where TP and FP denote the number of true and false positive samples, while P and N are the number of positive or negative macroblocks, respectively. In order to plot the ROC curve, the true positive rate is plotted against the false positive rate. Notice that the ROC curves characterize the separability between the conditional likelihoods of a feature $\mathbf{x}_i(t)$ under the two hypotheses $p(\mathbf{x}_i(t)|\mathcal{S}_i(t) = 1)$ and $p(\mathbf{x}_i(t)|\mathcal{S}_i(t) = 0)$. Therefore, they describe the intrinsic ability of a feature to discriminate between positive and negative samples.

We computed the ROC curves for three CIF resolution (352 × 288) video sequences: *News*, *Foreman* and *Mobile & Calendar*, encoded according to the coding conditions in Table I. The three videos are characterized by diverse motion and texture content, as illustrated in Table II, where SI and TI are spatial and temporal activity indexes computed as specified in [41]. Each coded slice is packetized according

TABLE II
CHARACTERISTICS OF THE TEST MATERIAL

Sequence	Resolution	bitrate [kbps]	PSNR [dB]	QP	SI	TI
<i>Foreman</i>	CIF 30fps	353	34.4	32	13.1	36.0
<i>News</i>	CIF 30fps	283	37.3	31	17.0	30.1
<i>Mobile</i>	CIF 30fps	532	28.3	36	24.5	37.0
<i>Paris</i>	CIF 30fps	480	33.6	32	19.5	16.4
<i>Mother</i>	CIF 30fps	150	37.0	32	8.5	9.5
<i>Crowdrun</i>	4CIF 25fps	6757	33.5	30	14.2	48.1
<i>Ducks</i>	4CIF 25fps	7851	30.4	34	16.2	43.3
<i>Harbour</i>	4CIF 30fps	5453	36.3	28	13.5	38.0

to the real-time transfer protocol (RTP) specifications [6]. The simulated error-prone channel drops coded packets according to a given packet loss rate (PLR). The error patterns have been generated using a two-state Gilbert's model [36]. The work in [42] considers a maximum burst length of 9 packets as characteristic of IP networks, with an average around 2-3 packets. In our experiments we tuned the model parameters to obtain an average burst length of three packets. For each considered PLR value, we simulated the transmission of the test sequences over 15 channel realizations. We employ the error concealment algorithm provided by the reference decoder [40], which implements a MCTC for P frames, and a SC with bilinear interpolation for I frames. The decay parameters $\alpha_{1,S}, \alpha_{0,S}, \beta_{1,S}, \beta_{0,S}, \alpha_{1,T}, \alpha_{0,T}, \beta_{1,T}, \beta_{0,T}$ are reported in Table III. In order to empirically determine the parameter values, we concatenated several video sequences, with different temporal and spatial characteristics, which were not included in the test material (*Hall Monitor, Container, Soccer, City, Ice, ParkJoy*). Then, we corrupted these sequences and we extracted the features to obtain histograms as those shown in Figure 4 and Figure 5. By fitting an exponential function to these histograms, we found the values of the parameters. We verified the sensitivity of the parameters to different coding conditions. We found that, aggregating the features extracted from diverse sequences, the parameters were only mildly dependent on the coding conditions, at least when the visual quality was not severely affected by aggressive lossy coding.

Figures 6 and 7 show the ROC curves for the features described in Section IV-B.1 and IV-B.2, respectively. We evaluated both the features $\mathbf{x}_{i,T}^A(t)$ and $\mathbf{x}_{i,S}^A(t)$ alone, and their combination with $\mathbf{x}_{i,T}^B(t)$ and $\mathbf{x}_{i,S}^B(t)$. To give a concise measure of the discriminative power of the feature, we also give the value of the area under the curve (AUC) in the figures. Values

TABLE III

DECAY PARAMETERS USED IN THE EXPERIMENTS FOR THE CONDITIONAL LIKELIHOOD FUNCTIONS.

Concealment type	Feature	$\mathcal{S}_i(t) = 1$	$\mathcal{S}_i(t) = 0$
Motion-compensated temporal concealment	$\mathbf{x}_{i,T}^A(t)$	$\alpha_{1,T} = 11$	$\alpha_{0,T} = 7$
	$\mathbf{x}_{i,T}^B(t)$	$\beta_{1,T} = 0.2$	$\beta_{0,T} = 0.3$
Spatial concealment	$\mathbf{x}_{i,S}^A(t)$	$\alpha_{1,S} = 0.02$	$\alpha_{0,S} = 0.01$
	$\mathbf{x}_{i,S}^B(t)$	$\beta_{1,S} = 0.01$	$\beta_{0,S} = 0.05$

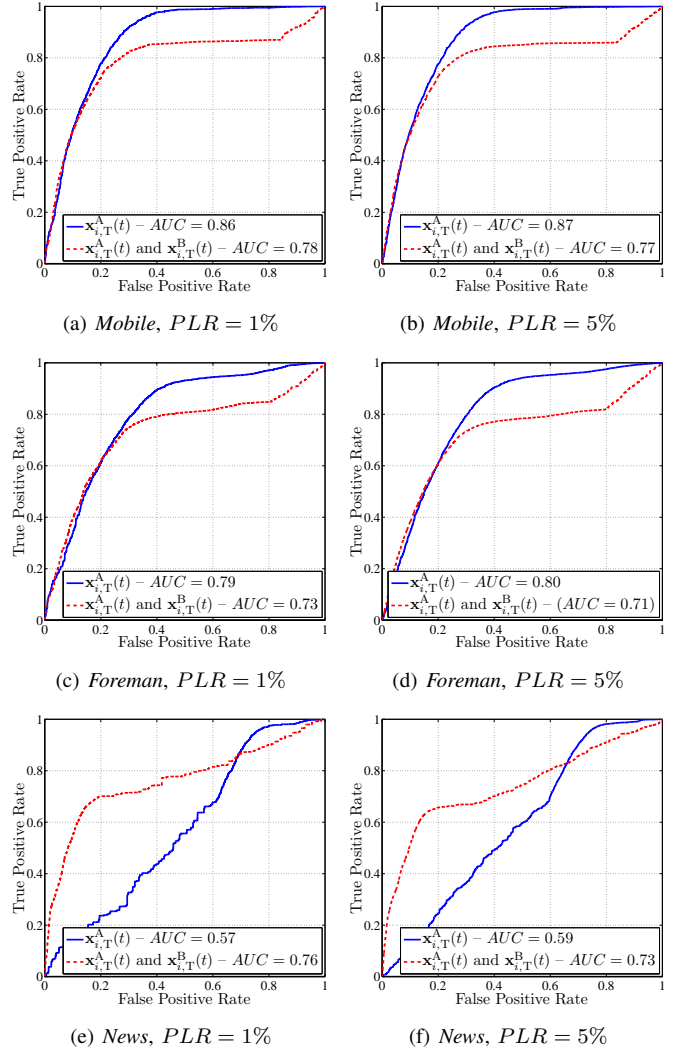


Fig. 6. ROC curves for the features adopted in the case of motion-compensated temporal concealment.

of the AUC close to one indicate better separability. Notice that the AUC values for spatial concealment are generally higher than for temporal concealment, as in the latter case errors are concealed better and, consequently, are more difficult to detect. We observe that the feature vector formed by $\mathbf{x}_{i,T}^A(t)$ and $\mathbf{x}_{i,T}^B(t)$ is less discriminative than $\mathbf{x}_{i,T}^A(t)$ alone, for some video sequences (namely, *Foreman* and *Mobile*). This is due to the variability of the motion field. Recall that, in order to compute $\mathbf{x}_{i,T}^B(t)$, we approximate the motion field in the current frame with the motion field in the previous frame. Such approximation is valid only if the motion field does not change too much from one frame to another. In order to quantify how rapidly the motion field changes, we compute the total motion difference (TMD) for each frame, defined as:

$$TMD(t) = \sum_{i=1}^N \left| \widehat{MV}_i(t) - \widehat{MV}_i(t-1) \right|. \quad (21)$$

As an example, Figure 8 shows the value of $TMD(t)$ for the first 100 frames of *Foreman*, *News* and *Mobile* sequences. The values of $TMD(t)$ of the *Foreman* sequence is higher, on

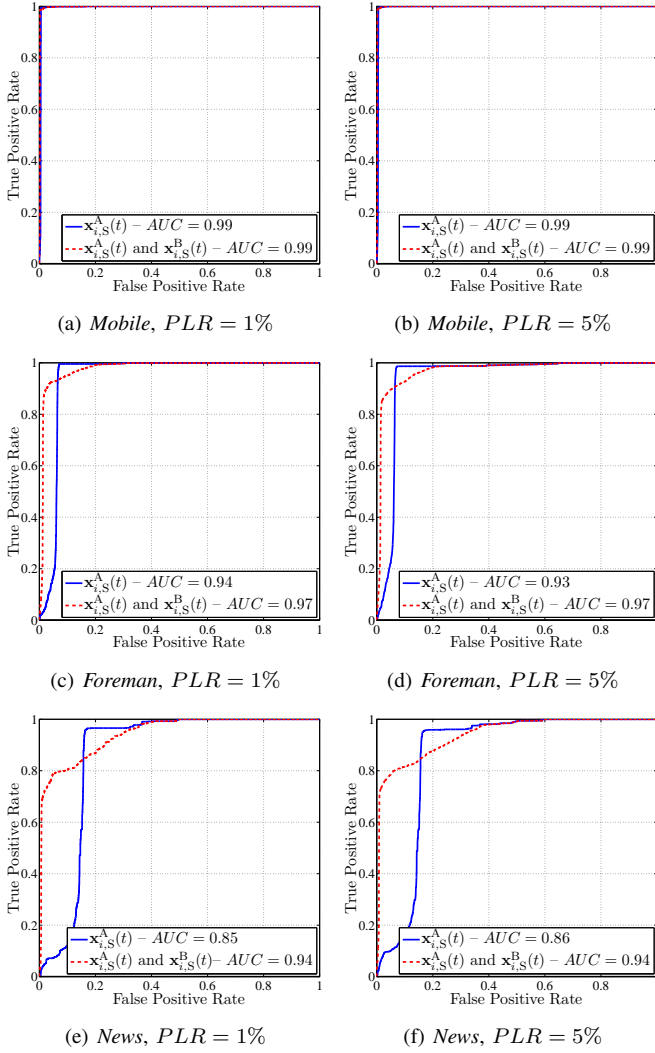


Fig. 7. ROC curves for the features adopted in the case of spatial concealment.

average, than the values of the other sequences. If $TMD(t)$ is too large, the estimate of $\mathbf{x}_{i,T}^B(t)$ obtained from the previous frame is practically meaningless. This explains why the estimate of $\mathbf{x}_{i,T}^B(t)$ obtained from the previous frame reduces the AUC in Figures 6(a)-(d). To prevent this problem, we compute $TMD(t)$ for each frame. For those frames where $TMD(t)$ is larger than a given threshold (set here to $4 \cdot 10^5$ using quarter-pel motion vectors), we consider only the $\mathbf{x}_{i,T}^A(t)$ feature.

D. Efficient MAP estimation using graph cuts

In order to find a maximum-a-posteriori estimate $\hat{S}(t)$, the log-posterior distribution (12) must be maximized. An exhaustive search among the 2^N possible values of $\ln p(S(t)|\mathbf{x}(t))$ is impractical, since N is of the order of $10^5 - 10^6$. Interestingly, Greig et al. [39] showed that the maximization of (12) has an equivalent formulation in terms of optimal flows/minimum cuts in a graph. Figure 9 illustrates this interpretation. Following [38], we represent each frame as a weighted directed graph in which each of the N macroblocks is a node. Furthermore, there are two additional nodes that represent

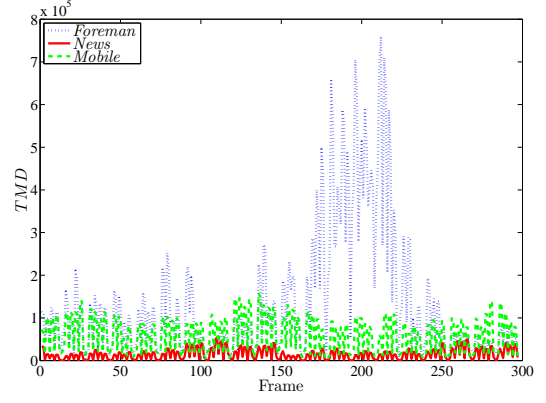


Fig. 8. $TMD(t)$ for each frame of the *Foreman*, *News* and *Mobile* sequences.

the two states, i.e. ‘lost’ and ‘received’. Similarly, there are two kinds of arcs: the so-called t-links, which connect each macroblock to a terminal node; and the n-links, which connect neighboring macroblocks according to a neighborhood lattice (e.g. 4-connectivity). The capacities of t-links correspond to the conditional log-likelihood of assigning the corresponding state to that macroblock, i.e. $\ln [p(\mathbf{x}_i(t)|\mathcal{S}_i(t) = 1)]$ or $\ln [p(\mathbf{x}_i(t)|\mathcal{S}_i(t) = 0)]$. The costs w_{ij} of n-links correspond to a penalty for discontinuity between the macroblocks, and are related to the prior term (interaction potential) in (12). A reasonable choice for the interaction potential is to assign weights to n-links proportionally to the difference of likelihood values [43]:

$$w_{ij} = k_{ij} \cdot |p(\mathbf{x}_i(t)|\mathcal{S}_i(t) = 1) - p(\mathbf{x}_j(t)|\mathcal{S}_j(t) = 1)|, \quad (22)$$

where the coefficient k_{ij} accounts for specific spatial correlation related, e.g., to a given slicing structure. This implies that likelihood differences may be penalized more in the horizontal or vertical direction. A *cut* in the graph of Figure 9 is a partition of the nodes into two disjoint sets, L and R , with the property that terminal nodes cannot be in the same partition. It can be shown that finding a cut with minimum cost is equivalent to obtain the MAP estimate $\hat{S}(t)$ [39]. Moreover, by the Ford-Fulkerson algorithm [44], finding a cut with minimum cost corresponds to maximizing the total flow from the source to the sink in the graph (i.e., between terminal nodes), and there are efficient algorithms able to solve this problem in $O(N)$ time.

Figure 10 shows the result of the graph cut algorithm applied on a P frame of the *Mobile* sequence. We use a neighborhood for the MRF composed of 4-connected macroblocks, as shown in Figure 9. The weights of the n-links k_{ij} have been set to 1 for the horizontal edges, and to 0.4 for the vertical ones. In this way, we tend to favor patterns of losses composed by horizontal slices. The lost macroblocks are highlighted in Figure 10(a). Instead, Figure 10(b) shows the likelihood function $p(\mathbf{x}_i(t)|\mathcal{S}_i(t) = 1)$, which roughly identifies potentially lost MBs. This noisy estimate is refined through the MRF prior using the graph cut method explained above, which produces as output the binary labeling shown in Figure 10(c). We used the graph cut implementation made

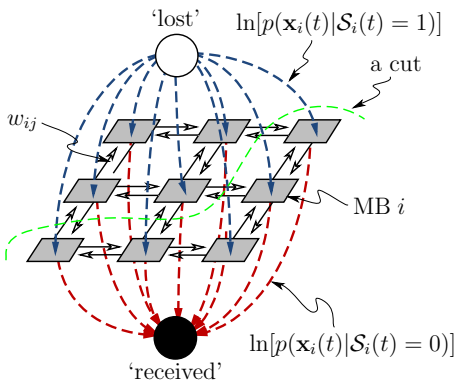
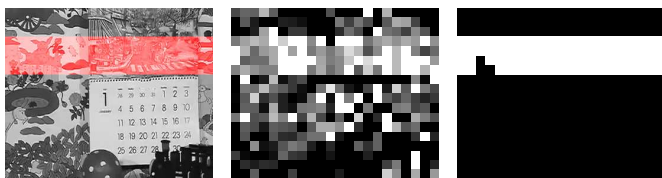


Fig. 9. Graph-cut representation of the MAP estimation of $\mathcal{S}(t)$ for a simple frame of 3×3 macroblocks. Each MB can be seen as a node in a graph with symmetric capacitated arcs ($w_{ij} = w_{ji}$). T-links connect each MB to the state of ‘lost’ or ‘received’, with weights equal to their conditional log-likelihoods. A cut corresponds to a partition of the nodes.



(a) Corrupted frame (b) Likelihood function (c) MAP $\hat{\mathcal{S}}$

Fig. 10. Likelihood and resulting MAP estimate, for temporal error concealment.

available by the authors of [38]. Notice how most of the false positives are canceled out in the MAP labeling.

As we did in Section IV-C, we evaluate the performance of the MAP labeling in terms of FPR and TPR. The results are reported in Table IV. We also compute the *accuracy* of the binary labeling, that is:

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN}. \quad (23)$$

The detection of the support of the initial error is generally better for spatial concealment. This is mainly due to the higher discrimination power of features for this kind of error concealment, as discussed in Section IV-C.

TABLE IV
LABELING PERFORMANCE OF THE MAP ESTIMATOR.

(a) $PLR = 1\%$						
	MCTC			SC		
	FPR	TPR	Acc.	FPR	TPR	Acc.
Foreman	0.33	0.89	0.87	0.05	0.97	0.95
News	0.12	0.62	0.81	0.14	0.96	0.86
Mobile	0.13	0.92	0.87	$5 \cdot 10^{-5}$	0.99	0.99
(b) $PLR = 5\%$						
	MCTC			SC		
	FPR	TPR	Acc.	FPR	TPR	Acc.
Foreman	0.33	0.90	0.88	0.05	0.97	0.95
News	0.12	0.60	0.82	0.14	0.96	0.87
Mobile	0.14	0.92	0.86	$4 \cdot 10^{-4}$	0.99	0.99

V. APPLICATION TO THE ESTIMATION OF CHANNEL-INDUCED DISTORTION

An accurate estimate $\hat{\mathcal{S}}$ of the support of the initial error is a valuable tool in no-reference pixel quality assessment. Indeed, this estimate can be used as input to other no-reference or reduced-reference video quality monitoring systems, which cannot be employed when the bitstream is not available. As a proof of concept, we demonstrate the applicability of our estimation method by describing how $\hat{\mathcal{S}}$ can be plugged into a NR-BP system — namely, the NORM algorithm described in [21] — that estimates the channel-induced MSE distortion.

A. Extending the hybrid NORM system to the NR-P scenario

The NO-Reference quality Monitoring (NORM) algorithm [21] is a NR-BP method designed to estimate the channel-induced MSE distortion $\hat{d}_i(t)$ that relies on both the received bitstream and the decoded pixels. The algorithm is conceived to work with a H.264/AVC bitstream and the default error concealment implemented in the JM reference software [40], but the very same principles can be easily extended to other coding standards and concealment methods. NORM takes as input the decoded frame, the received/concealed motion vectors, prediction residuals and coding modes, as well as the pattern of channel errors \mathcal{I} , which is immediately available from the bitstream. To estimate channel distortion, NORM first estimates the initial error $d_i^I(t)$ in (3), produced whenever error concealment fails to adequately recover the original frame content. The NORM algorithm estimates then the propagation of the distortion due to prediction, i.e. the $d_i^P(t_1, \dots, t_n)$ in (3). Finally, an estimate of the distortion $\hat{d}_i(t)$ is obtained recursively through (4). We refer the interested reader to [21] for further details on the estimation of the initial error and of distortion propagation.

In the NR-P scenario, the only information available is the decoded video \hat{X} . Thus, in order to run NORM, a set of encoding parameters such as motion vectors, prediction residuals and coding modes are to be estimated from the decoded sequence, together with the support of the initial error \mathcal{S} as detailed in the previous section. In order to estimate motion vectors and prediction residuals, motion estimation has to be performed at the decoder side. Generally speaking, it is not possible to find exactly the original motion field coded in the bitstream without relying on a priori assumptions (e.g., assuming that each macroblock in a frame adopts the same quantization parameter [45]). However, in order to find the MSE distortion $\hat{d}_i(t)$ using NORM, it is not necessary to reconstruct the original motion field. Indeed, we tested the sensitivity of the NORM algorithm to different motion estimation conditions, by feeding NORM with motion vectors estimated on the decoded (corrupted) video. We computed the linear correlation coefficient between the MSE distortion estimated by NORM with the original motion vectors contained in the bitstream, and the one obtained using the motion vectors estimated at the decoder. In our experiments, we observe that there is basically no loss in estimating the MSE distortion, using 4×4 sub-block partitions and sub-pixel motion refinement (correlation

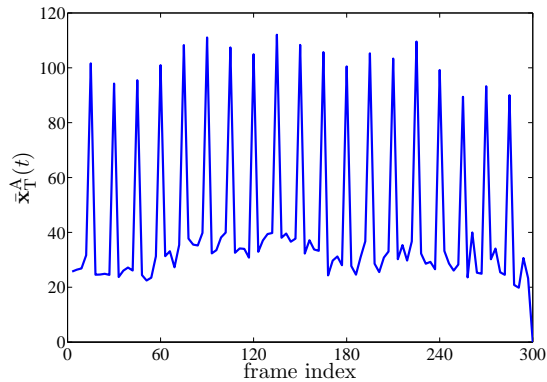


Fig. 11. Average temporal prediction error with motion vectors estimated at the decoder side, for the *Mobile* sequence. The Intra-refresh period of 15 frames can be easily inferred from the peaks in the graph.

coefficient with NORM’s distortion larger the 0.98 at the frame level).

Once motion vectors have been found, prediction residuals are readily obtained. As for coding modes (SKIP, INTRA or INTER with different sub-block size, etc.), it is not possible, in general, to recover them from the decoded video. However, for our purpose we only need to distinguish between INTRA and INTER predicted blocks, which undergo a different error-concealment strategy. In [21], it is shown that losing the coding mode information about INTRA coded blocks in P slices does not affect substantially the prediction performance, as the average percentage of this type of blocks is usually below 4%. Since spatial error concealment can be used in IDR (Instantaneous Decoding Refresh) or I (Intra) pictures, we focus on detecting this kind of pictures. This turns out to be relatively simple, as in these frames no kind of temporal prediction is used. In fact, blocks in an I frame are predicted from spatial neighbors. Thus it is very unlikely that the reconstructed pixels are equal to pixels in a previous temporally predicted frame. In order to detect which frames are I/IDR, we can compute the average temporal prediction residual energy defined in (14), i.e., $\bar{x}_T^A(t) = \frac{1}{N} \sum_{i=1}^N x_{i,T}^A(t)$. In I/IDR pictures it will be much larger than in P/B frames. Figure 11 illustrates well this behavior: I frames correspond to well-discernible peaks in the energy of the prediction residuals.

B. Evaluation of the NR-P distortion estimation

In this section we evaluate the accuracy of the channel-induced distortion $\hat{d}(t)$ estimated by adapting the NORM algorithm to the NR-P scenario, as described in Section V-A. To this end, we corrupted five video sequences at CIF (352×288) spatial resolution, namely *Foreman*, *News*, *Mobile*, *Paris* and *Mother*, and three sequences at 4CIF (704×576) resolution (*Crowdrun*, *Ducks* and *Harbour*). A concise description of the test material and coding parameters is provided in Table I and Table II

We simulated the transmission of each video sequence over a lossy channel using Gilbert’s packet loss model [36], with an average burst length of three packets. Specifically, we generated two channel realizations for each PLR in the

set $\{0.1\%, 0.4\%, 1\%, 3\%, 5\%, 10\%\}$. Then, we estimated the MSE distortion in no-reference mode using: i) the original NR-BP NORM algorithm, and ii) the proposed NR-P MAP method, which leverages the estimated packet loss map $\hat{S}_i(t)$.

Figure 12 and Figure 13 visually compare the frame-level MSE distortion estimated by both NORM and NR-P MAP for a subset of the test sequences, with respect to the ground-truth distortion computed in full-reference mode. In these figures, each data point corresponds to one frame. For each test sequence, we computed Pearson’s linear correlation coefficients ρ between the no-reference and full-reference MSE distortion. Table V reports the corresponding values of ρ at the frame-level (column ‘Frm’) for both methods, obtained aggregating all channel realizations at different PLRs, for each of the test sequences. The value of the correlation coefficient is always higher than 0.82 for NR-P MAP and 0.93 for NORM. In addition, we computed ρ by aggregating all test sequences together, in order to demonstrate the content-independence of the tested methods. In this case, the correlation coefficient is, respectively, 0.90 for NR-P MAP and 0.98 for NORM. We notice that for the 4CIF resolution video sequences the spread of the estimated MSE values is generally larger than for the CIF sequences. This can be visually appreciated in Figures 12 and 13, where we also report the root mean square error (RMSE) measure with respect to a linear fit of the data. In the figures it can be noticed that the same relationship also holds for the NR-BP NORM method, which represents a sort of upper bound on the performance of the proposed NR-P method. The different estimation accuracy at the frame level with respect to CIF resolution can be justified by observing that, in the test material, the TI index for the 4CIF sequences is larger than for CIF sequences, as shown in Table II. This is aligned with the published results obtained for NORM [21], for which it was shown that the MSE estimation accuracy is generally lower in the case of video characterized by complex motion.

In many application scenarios, it is useful to produce a concise measure of the MSE distortion at the sequence level. To this end, we pooled together the MSE values obtained at the frame level, by temporal averaging across all the frames in each sequence. In this case, the impact of outliers, i.e. those frames for which the estimated MSE differs from the true MSE, is reduced. As a result, higher values of Pearson’s correlation coefficient are obtained, as shown in Table V (column ‘Seq’), i.e., 0.98 for NR-P MAP and 0.99 for NORM. Based on this result, we claim that the MSE distortion estimated by NR-P MAP is a very good approximation of the MSE estimated by NORM in a NR-BP setting. Further, it provides a reliable estimate of the ground-truth MSE computed in full-reference mode. This is illustrated in Figure 14, where we compare the MSE distortion computed either in no-reference or full-reference mode. Each data point corresponds to a corrupted sequence subject to a distinct channel realization. We included all eight test sequences and all realizations at different PLRs

It is widely acknowledged that, in general, MSE is not well correlated with human perception, as assessed by means of MOS values [46]. This is mainly due to the wide variety of visual impairments that can affect the quality of video sequences.

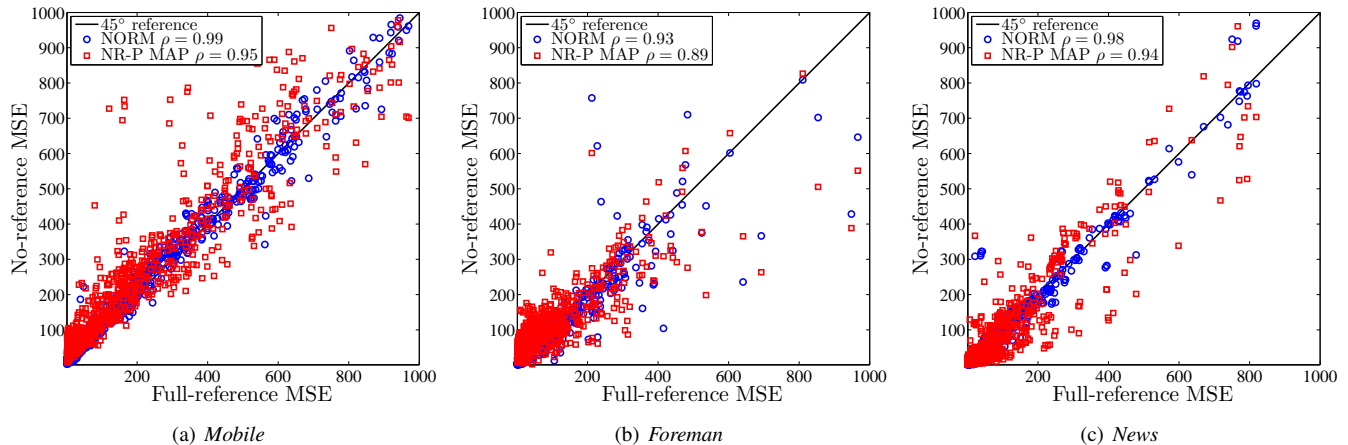


Fig. 12. Frame-level scatter plots for three CIF sequences. The RMSE values are: 82.47 (*Mobile*); 43.6 (*Foreman*); and 37.21 (*News*).

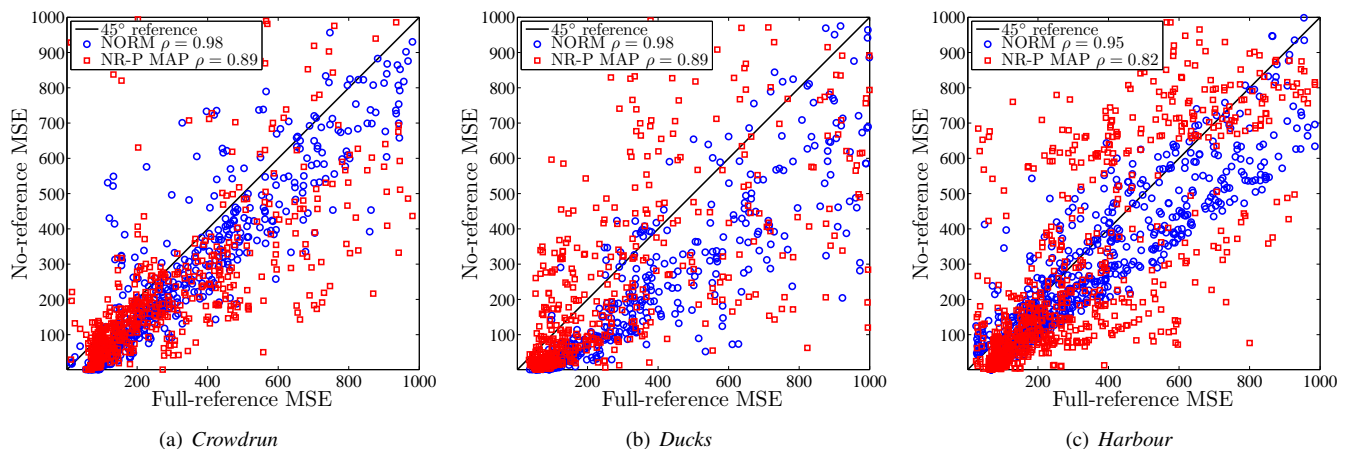


Fig. 13. Frame-level scatter plots for three 4CIF sequences. The RMSE values are: 123.02 (*Crowdrun*); 170.43 (*Ducks*); and 122.77 (*Harbour*).

In practice, when considering visual quality, it is customary to consider, in place of the MSE, its logarithmic version, i.e. PSNR, since it takes better into account the non-linear response of the human visual system. In [31] it has been shown that, for the specific case of channel-induced distortion, PSNR is relatively well correlated to perceptual quality as judged by human observers. Therefore, we map the MSE estimated with our approach to PSNR values, and compare the result with the subjective scores publicly available in the EPFL-PoliMI video quality assessment database [47]. The video sequences and channel realizations in [47] correspond to those used in our tests. Figure 15 shows the correlation between MOS values and estimated PSNR, when either NORM or NR-P MAP are used. We show separate charts for sequences at CIF and 4CIF spatial resolution. Indeed, in [47], different viewing distances were used depending on the spatial resolution, and raw subjective scores were normalized separately in order to remove outliers and subject bias.

We computed Pearson’s linear correlation coefficient obtaining 0.77 at CIF resolution and 0.94 at 4CIF resolution. We notice that, in both cases, the proposed NR-P MAP method approximates NORM (and, consequently, full-reference PSNR) in terms of correlation with MOS. The lower value of ρ for CIF

TABLE V
PEARSON’S LINEAR CORRELATION COEFFICIENT BETWEEN \hat{d} AND GROUNDTRUTH DISTORTION AT FRAME (FRM) AND SEQUENCE (SEQ) LEVEL.

Sequence	NR-P MAP		NORM	
	Frm	Seq	Frm	Seq
<i>Foreman</i>	0.89	0.98	0.93	0.99
<i>News</i>	0.94	0.97	0.98	0.99
<i>Mobile</i>	0.95	0.99	0.99	0.99
<i>Paris</i>	0.83	0.99	0.94	0.99
<i>Mother</i>	0.85	0.97	0.98	0.99
<i>Crowdrun</i>	0.89	0.99	0.98	0.99
<i>Ducks</i>	0.89	0.99	0.98	0.99
<i>Harbour</i>	0.82	0.98	0.95	0.98
Tot	0.90	0.98	0.98	0.99

sequences is likely to be attributed to the higher diversity in the spatio-temporal characteristics of the test material. Indeed, we observed the same behavior using NORM, which is a close approximation of full-reference PSNR.

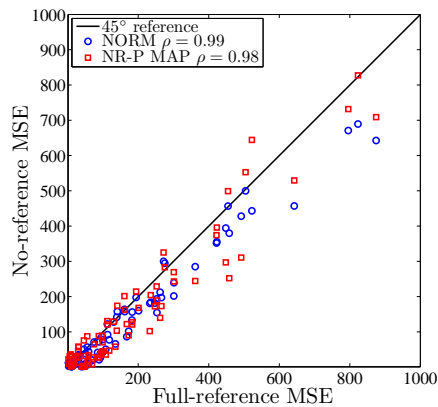
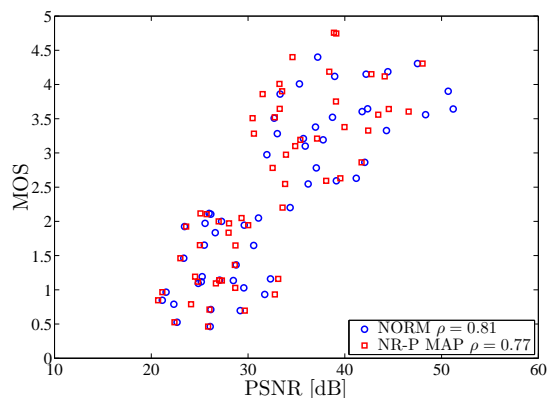
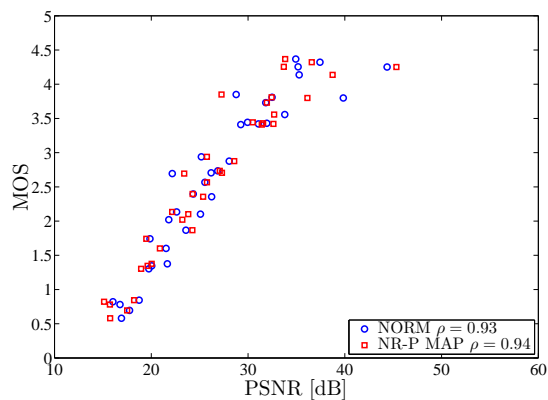


Fig. 14. Sequence-level scatter plot for all the eight tested sequences.



(a) CIF



(b) 4CIF

Fig. 15. Mean Opinion Score vs. PSNR estimated in no-reference mode.

Of course, further improvements can be pursued by leveraging MSE distortion measures obtained either in full-reference, NR-BP or NR-P mode, in more sophisticated ways, i.e.: i) incorporating non-uniform spatio-temporal pooling; ii) embedding the spatio-temporal masking effects related to characteristics of the video content; iii) identifying a suitable non-linear model that maps the observed distortion into predicted MOS values. These explorations are left to future work.

VI. CONCLUSIONS AND FUTURE RESEARCH DIRECTIONS

In this paper we present a video quality monitoring system tailored to visual impairments due to channel errors. Our approach assumes only the knowledge of the (corrupted) decoded pixels, and differs from other no-reference methods described in the literature in that it does assume only the knowledge of the decoded pixels. Therefore, it can be applied even when the bitstream information is missing, e.g. because it is encrypted. We show that: i) it is possible to infer the pattern of losses from the decoded pixels only; and ii) that this information enables to compute accurate estimates of the MSE distortion and of the subjective quality of a video sequence. Future work includes improving the estimation of the support of the initial error — e.g., by embedding the temporal correlation of errors due to the bursty nature of channel losses; or considering new feature such as blockiness at macroblocks boundaries left by spatial concealment. Separating channel-induced artifacts from characteristics of the video itself still remains the most difficult part in NR-P quality estimation. Finding such features has actually a deep impact on fields other than quality assessment, such as video forensics, or video restoration, as it enables to trace back the degradation undergone by the video content.

REFERENCES

- [1] G. Valenzise, S. Magni, M. Tagliasacchi, and S. Tubaro, “Estimating channel-induced distortion in H.264/AVC video without bitstream information,” in *Proc. 2nd Int. Workshop on Quality of Multimedia Experience*, Trondheim, Norway, June 2010.
- [2] M. Pinson and S. Wolf, “Low bandwidth reduced reference video quality monitoring system,” in *First International Workshop on Video Processing and Quality Metrics for Consumer Electronics*, Scottsdale, AZ, USA, 2005.
- [3] T. Yamada, Y. Miyamoto, M. Serizawa, and H. Harasaki, “Reduced-reference based video quality-metrics using representative-luminance values,” in *Third International Workshop on Video Processing and Quality Metrics for Consumer Electronics*, Scottsdale, AZ, USA, 2007.
- [4] G. Valenzise, M. Naccari, M. Tagliasacchi, and S. Tubaro, “Reduced-reference estimation of channel-induced video distortion using distributed source coding,” in *Proc. ACM Int. Conf. on Multimedia*, Vancouver, Canada, 2008.
- [5] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, “Overview of the H.264/AVC video coding standard,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, July 2003.
- [6] S. Wenger, “H. 264/AVC over IP,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 645–656, 2003.
- [7] T. Stockhammer, M.M. Hannuksela, and T. Wiegand, “H.264/avc in wireless environments,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 657–673, 2003.
- [8] H. Liu and I. Heynderickx, “A no-reference perceptual blockiness metric,” in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Las Vegas, NV, USA, April 2008, pp. 865–868.
- [9] P. Marziliano, F. Dufaux, S. Winkler, and Ebrahimi, “A no-reference perceptual blur metric,” in *Proc. IEEE Int. Conf. Image Processing*, Rochester, NY, USA, September 2002, vol. 3, pp. 57–60.
- [10] A. Leontaris and A.R. Reibman, “Comparison of blocking and blurring metrics for video compression,” in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Philadelphia, PA, USA, March 2005, pp. 585–588.
- [11] A.R. Reibman, S. Sen, and J. Van der Merwe, “Analyzing the spatial quality of Internet streaming video,” in *Proc. Workshop on Video Processing and Quality Metrics for Consumer Electronics*, Scottsdale, AZ, USA, January 2005.
- [12] A. Leontaris, P.C. Cosman, and A.R. Reibman, “Quality evaluation of motion-compensated edge artifacts in compressed video,” *IEEE Trans. Image Process.*, vol. 16, no. 4, pp. 943–956, 2007.
- [13] A. Ichigaya, M. Kurozumi, N. Hara, Y. Nishida, and E. Nakasu, “A method of estimating coding PSNR using quantized DCT coefficients,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 2, pp. 251–259, February 2006.

- [14] A. Ichigaya, Y. Nishida, and E. Nakasu, "Nonreference method for estimating PSNR of MPEG-2 coded video by using DCT coefficients and picture energy," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 6, pp. 817–826, June 2008.
- [15] T. Brandão and M. P. Queluz, "Blind PSNR estimation of video sequences using quantized DCT coefficient data," in *Picture Coding Symposium*, Lisbon, Portugal, November 2007.
- [16] T. Brandão and M.P. Queluz, "No-reference quality assessment of H.264/AVC encoded video," *IEEE Trans. Circuits Syst. Video Techn.*, vol. 20, no. 11, pp. 1437–1447, 2010.
- [17] A. R. Reibman, V. A. Vaishmpayan, and Y. Sermadevi, "Quality monitoring of video over a packet network," *IEEE Trans. Multimedia*, vol. 6, no. 2, pp. 327–334, April 2004.
- [18] T. Yamada, S. Yachida, Y. Senda, and M. Serizawa, "Accurate video-quality estimation without video decoding," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Dallas, TX, USA, March 2010.
- [19] A.R. Reibman and D. Poole, "Characterizing packet-loss impairments in compressed video," in *Proc. IEEE Int. Conf. Image Processing*, San Antonio, TX, USA, September 2007, vol. 5.
- [20] T.L. Lin, S. Kanumuri, Y. Zhi, D. Poole, P.C. Cosman, and A.R. Reibman, "A versatile model for packet loss visibility and its application to packet prioritization," *IEEE Transactions on Image Processing*, vol. 19, no. 3, pp. 722–735, March 2010.
- [21] M. Naccari, M. Tagliasacchi, and S. Tubaro, "No-reference video quality monitoring for H.264/AVC coded video," *IEEE Trans. Multimedia*, vol. 11, no. 5, pp. 932–946, August 2009.
- [22] A. Raake, M.N. Garcia, S. Moller, J. Berger, F. Kling, P. List, J. Johann, and C. Heidemann, "T-V-model: Parameter-based prediction of IPTV quality," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Las Vegas, NV, USA, March 2008, pp. 1149–1152.
- [23] S. Winkler and F. Dufaux, "Video quality evaluation for mobile applications," in *Visual Communications and Image Processing, Proc of SPIE*, Lugano, Switzerland, 2003, vol. 5150, pp. 593–603.
- [24] R.R. Pastrana-Vidal and J.C. Gicquel, "Automatic quality assessment of video fluidity impairments using a no-reference metric," in *Proc. of Int. Workshop on Video Processing and Quality Metrics for Consumer Electronics*, Phoenix, AZ, USA, January 2006.
- [25] T. Liu, Y. Wang, J.M. Boyce, Z. Wu, and H. Yang, "Subjective quality evaluation of decoded video in the presence of packet losses," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Honolulu, Hawaii, April 2007, vol. 1.
- [26] F. Yang, S. Wan, Q. Xie, and H.R. Wu, "No-reference quality assessment for networked video via primary analysis of bit stream," *IEEE Trans. Circuits Syst. Video Techn.*, vol. 20, no. 11, pp. 1544–1554, 2010.
- [27] C. Yim and A.C. Bovik, "Evaluation of temporal variation of video quality in packet loss networks," *Sig. Proc.: Image Comm.*, vol. 26, no. 1, pp. 24–38, 2011.
- [28] J. Gustafsson, G. Heikkila, and M. Pettersson, "Measuring multimedia quality in mobile networks with an objective parametric model," in *Proc. IEEE Int. Conf. Image Processing*, San Diego, CA, USA, October 2008.
- [29] R.V. Babu, A.S. Bopardikar, A. Perkis, and O.I. Hillestad, "No-Reference metrics for video streaming applications," in *Proc. Int. Workshop on Packet Video*, Irvine, CA, USA, December 2004.
- [30] T. Yamada, Y. Miyamoto, and M. Serizawa, "No-reference video quality estimation based on error-concealment effectiveness," in *IEEE Packet Video*, Lausanne, Switzerland, November 2007.
- [31] M. Naccari, M. Tagliasacchi, and S. Tubaro, "Subjective evaluation of a no-reference video quality monitoring algorithm for H.264/AVC video over a noisy channel," in *Proc. Int. Conf. Image Processing*, Cairo, Egypt, November 2009.
- [32] S. Kanumuri, P.C. Cosman, A.R. Reibman, and V.A. Vaishmpayan, "Modeling packet-loss visibility in MPEG-2 video," *IEEE Trans. Multimedia*, vol. 8, no. 2, pp. 341, 2006.
- [33] A.R. Reibman and D. Poole, "Predicting packet-loss visibility using scene characteristics," in *Proc. Packet Video*, Lausanne, Switzerland, November 2007.
- [34] S. Kanumuri, S.G. Subramanian, P.C. Cosman, and A.R. Reibman, "Predicting H.264 packet loss visibility using a generalized linear model," in *Proc. IEEE Int. Conf. Image Processing*, Atlanta, GA, USA, October 2006, pp. 2245–2248.
- [35] N. Färber, K. Stuhlmüller, and B. Girod, "Analysis of error propagation in hybrid video coding with application to error resilience," in *Proc. IEEE Int. Conf. Image Processing*, Kobe, Japan, October 1999.
- [36] E. N. Gilbert, "Capacity of a burst-noise channel," *Bell System Technical Journal*, vol. 39, pp. 1253–1266, September 1960.
- [37] C.M. Bishop, *Pattern Recognition and Machine Learning*, chapter 8: Graphical Models, Springer-Verlag Inc., New York, NJ, USA, 2006.
- [38] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1124–1137, September 2004.
- [39] D.M. Greig, B.T. Porteous, and A.H. Seheult, "Exact maximum a posteriori estimation for binary images," *Journal of the Royal Statistical Society*, vol. 51, no. 2, pp. 271–279, 1989.
- [40] G. J. Sullivan, T. Wiegand, and K.-P. Lim, "Joint model reference encoding methods and decoding concealment methods," Tech. Rep. JVT-1049, Joint Video Team (JVT), September 2003.
- [41] ITU-T, *Recommendation ITU-R P 910*, September 1999, Subjective video quality assessment methods for multimedia applications.
- [42] T.-K. Chua and D. C. Pheanis, "QoS evaluation of sender-based loss-recovery techniques for VoIP," *IEEE Netw.*, vol. 20, no. 6, pp. 14–22, December 2006.
- [43] Y. Boykov and G. Funka-Lea, "Graph cuts and efficient N-D image segmentation," *International Journal of Computer Vision*, vol. 70, no. 2, pp. 109–131, 2006.
- [44] L.R. Ford and D.R. Fulkerson, *Flows in networks*, Princeton University Press, 1962.
- [45] G. Valenzise, M. Tagliasacchi, and S. Tubaro, "Estimating QP and motion vectors in H.264/AVC video from decoded pixels," in *Proc. ACM Int. Workshop on Multimedia in Forensics, Security and Intelligence*, Firenze, Italy, October 2010.
- [46] B. Girod, "What's wrong with mean-squared error?," *MIT Press Cambridge, MA, USA*, 1993.
- [47] F. De Simone, M. Tagliasacchi, M. Naccari, S. Tubaro, and T. Ebrahimi, "A H.264/AVC video database for the evaluation of quality metrics," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, March 2010, pp. 2430–2433.