No-Reference Video Quality Monitoring for H.264/AVC Coded Video

Matteo Naccari, *Student Member, IEEE*, Marco Tagliasacchi, *Member, IEEE*, and Stefano Tubaro, *Member, IEEE*

***Abstract*—When video is transmitted over a packet-switched network, the sequence reconstructed at the receiver side might suffer from impairments introduced by packet losses, which can only be partially healed by the action of error concealment techniques. In this context we propose NORM (NO-Reference video quality Monitoring), an algorithm to assess the quality degradation of H.264/AVC video affected by channel errors. NORM works at the receiver side where both the original and the uncorrupted video content is unavailable. We explicitly account for distortion introduced by spatial and temporal error concealment together with the effect of temporal motion-compensation. NORM provides an estimate of the mean square error distortion at the macroblock level, showing good linear correlation (correlation coefficient greater than 0.80) with the distortion computed in full-reference mode. In addition, the estimate at the macroblock level can be successfully exploited by forward quality monitoring systems that compute quality objective metrics to predict mean opinion score (MOS) values. As a proof of concept, we feed the output of NORM to a reduced-reference quality monitoring system that computes an estimate of the structural similarity metric (SSIM) score, which is known to be well correlated with perceptual quality.**

***Index Terms*—Video coding, video quality monitoring.**

I. Introduction

THE use of IP networks for the delivery of multimedia contents is gaining an increasing popularity as a mean of broadcasting media files from a content provider to many content consumers. In the case of video, for instance, packet-switched networks are used to distribute programs in IPTV applications. Typically, these kinds of networks provide only best-effort services, i.e., there is no guarantee that the content will be delivered without errors to the final users.

In some circumstances, the content provider and the user might decide to stipulate a service level agreement (SLA) that fixes an expected perceived quality at the end-user terminal: the provider fixes a price to the customers for assuring the agreed quality-of-service (QoS), and pays a penalty if the SLA is unfulfilled. For this reason, it is fundamental in IP networks in particular, and video broadcasting applications in general, to assess the visual quality of distributed video contents.

Manuscript received October 01, 2008; revised March 10, 2009. First published May 02, 2009; current version published July 17, 2009. This work was presented in part in [1] and has been developed within VISNET II, a European Network of Excellence (http://www.visnet-noe.org), funded under the European Commission IST FP6 programme. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Thinh Nguyen.

The authors are with the Dipartimento di Elettronica e Informazione, Politecnico di Milano, 20133 Milano, Italy (e-mail: matteo.naccari@polimi.it; marco.tagliasacchi@polimi.it; stefano.tubaro@polimi.it).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2009.2021785

In practice, the received video sequence may be a degraded version of the original one. Besides the distortion introduced by lossy coding, the user's experience might be affected by channel induced distortions. In fact, the channel might drop packets, thus introducing errors that propagate along the decoded video content because of the predictive nature of conventional video coding schemes [2]–[4], or it might cause jitter delay, due to decoder buffer underflows determined by network latencies. Although both aspects are equally important in determining the perceived quality, in this paper we focus on the former and we refer the reader to the available literature [5], [6] for aspects related to the effect of delay.

Video quality assessment can be realized by means of subjective or objective tests. In subjective tests, a group of people is asked to watch a set of video clips and rate their quality to obtain mean opinion scores (MOS). Subjective tests cannot be used for permanent in-service quality assessment applications due to the direct involvement of humans. Therefore, the overall video quality assessment must be performed by automated procedures. In objective tests, the quality of a video sequence is evaluated with metrics that aim to predict the MOS value on the basis of some features extracted from decoded videos [7], [8]. Objective quality metrics can be further classified in: full-reference, reduced-reference and no-reference, based on the amount of information available for comparison with the original content. Full-reference metrics require the entire reference content to be available at the receiver side: this constraint yields the full-reference metrics to be impractical for quality assessment of video content transmitted over a bandlimited noisy channel. Reduced-reference metrics rely on a coarse, feature-based representation of the transmitted video which is available at the receiver side without errors. No-reference objective metrics estimate the received video quality without any information on the error-free video content. Therefore, these metrics must rely on some assumptions as, for example, the type of distortion introduced in transmitted videos.

In this paper we propose NORM, a *NO-Reference video quality Monitoring* algorithm to automatically assess the channel induced distortion in a video sequence decoded from a H.264/AVC compliant bitstream, which has been transmitted through a noisy channel affected by packet losses. We explicitly model the effects of spatial and temporal error concealment, the loss of prediction residuals and the temporal distortion propagation due to the motion-compensation loop. Moreover, we analyze the contributions to the channel induced distortion introduced by the coding tools specific of the H.264/AVC video coding standard: intra-macroblock prediction and deblocking filter [9]. Despite previous works that present results of the estimated distortion at the frame and sequence level [10], NORM provides a reasonably accurate estimate of the MSE

distortion also at the macroblock (MB) level. In fact, recent studies [11] have shown that applying a region-of-interest weighting to the MSE computed at a local scale (e.g., macroblock) may considerably improve the fidelity of the quality assessment at the frame or sequence level, which is of most interest in user-oriented applications. Therefore, the estimate provided by the proposed algorithm can feed forward quality assessment systems that compute other objective metrics highly correlated with the MOS. As a concrete example to support this claim, we show that the output of the proposed no-reference distortion estimation algorithm can be efficiently exploited in a reduced-reference system to compute an approximation of the SSIM score (structural similarity metrics) [12], which has been proved to be strongly correlated with the MOS at the sequence level.

The remainder of this paper is organized as follows: Section II reviews the literature on channel induced distortion estimation by means of objective metrics. Section III presents the overall architecture of NORM as well as the algorithmic details for the computation of the individual terms contributing to the channel induced distortion. Section IV illustrates a reduced-reference system that exploits the distortion estimated by NORM at the macroblock level to compute the SSIM objective metrics. The experimental tests and the comparisons conducted to validate the proposed scheme are reported in Section V while Section VI concludes the paper.

II. Related Work

The estimation of the channel induced distortion of video transmitted over a noisy channel can be performed either at the transmitter or at the receiver side. At the transmitter, one has both the original and the encoded video sequence available, whereas the actual error pattern is unknown. Therefore the proposed techniques rely on a statistical representation of the channel and provide an estimate of the expected distortion at the frame [3], [2], macroblock or pixel [13], [14] level, where the expectation is taken over different realizations of the channel. In this scenario, the goal is to provide a mean of tuning encoder parameters to achieve optimal end-to-end coding efficiency. Conversely, at the receiver side, distortion estimation is somewhat simplified by the deterministic knowledge of the actual error pattern. However, the unavailability of the original video sequence, which is typical in practical applications, complicates the estimation of the distortion, which must be carried out in no-reference mode. Also, the methods proposed in the literature differ depending on the computational resources available at the receiver side.

The work in [10] describes an algorithm to estimate the MSE distortion for bitstreams coded with any conventional motion-compensated video codec. The estimation is provided at different levels of granularity and in three versions: Full Parse (FP), Quick Parse (QP), and No Parse (NP), each targeting a different tradeoff between computational complexity and estimation accuracy. The FP method achieves the highest accuracy and provides an estimate of the channel induced distortion at the pixel level. It requires to retrieve some parameters by entropy decoding and inverse quantization of the bitstream. The QP method provides the distortion at the slice level. The parameters needed in the QP estimation can be obtained by simply analyzing the received bitstream at the transport level and therefore without requiring further bitstream decoding. Finally the NP method does not perform any bitstream analysis but simply estimates the received video quality on the basis of the experienced packet loss rate (PLR) at the receiver side. Due to the limited computational complexity, these methods are particularly suitable for video quality monitoring by the network service provider point of view.

The system proposed in [15] aims at assessing the received video quality at the sequence level. A model is designed to tradeoff estimation accuracy, computational complexity and suitability for large scale video quality monitoring. First the authors derive a model to estimate the received video quality taking into account parameters such as the used codec, the adopted error concealment strategy, the bit-rate and the packetization used. Then, to avoid the dependence on the specific sequence, they introduce the relative PSNR (rPSNR) metrics, which is defined as the difference between the experienced PSNR at the receiver side and a given pre-negotiated target PSNR. The approach is validated on real video sequences and network conditions and the results reveal a good correlation between the predicted and the actual values.

The recent work in [16] proposes a no-reference MSE estimation algorithm that can be embedded in any H.264/AVC compliant decoder with only little computational complexity overhead. The method relies on the concept of error concealment effectiveness. Indeed, the authors observe that for some lost macroblocks the error concealment algorithm performs poorly due to several factors, e.g., motion complexity and local texturing of the lost macroblock. Therefore, they propose a heuristics that measures the error concealment effectiveness on the basis of motion information and boundary distortion between the lost macroblock and its neighbors. At the sequence level, there is a reasonably accurate quadratic fitting between the number of macroblocks judged as ineffectively error concealed and the MSE measured in full-reference mode.

A different approach is pursued in [17], where machine learning classifiers are used to predict packet loss visibility in MPEG-2 coded bitstreams. Training is performed on data collected by means of extensive subjective campaigns. Both a no-reference and a reduced-reference quality assessment algorithm are proposed. This methodology has been further adapted to H.264/AVC coded bitstreams in [18].

III. No-Reference Channel Induced Distortion Estimation Algorithm

A. Overall Setup

This section introduces the proposed NORM algorithm that computes an estimate of the channel induced distortion at macroblock level. NORM receives in input a H.264/AVC compliant bitstream that has been transmitted over a noisy channel. With reference to Fig. 1, the received bitstream is processed by the H.264/AVC decoder, which applies its own embedded concealment strategy over lost data. The decoded frame, together with the received/concealed motion vectors, prediction residuals and coding modes are fed into the proposed algorithm, which provides an estimate of the channel induced distortion $\hat{D}_n^i$, for the ith macroblock in frame n. The accuracy of this estimate can be evaluated at the macroblock, slice, frame and sequence granularity with respect to the actual distortion computed in full-reference mode, by comparing the decoded frame $\tilde{X}$ with the error free decoded frame $\hat{X}$. We notice that we explicitly consider

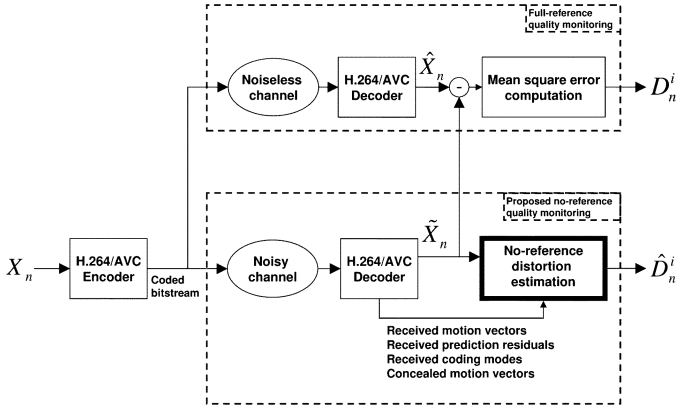

Fig. 1. Block diagram of the proposed NORM algorithm.

only the distortion due to channel losses. Conversely, the distortion introduced by lossy coding can be estimated using methods already available in the literature [19], [20]. Since these two terms can be approximately considered to be uncorrelated [21], they can be added together to obtain the overall distortion with respect to the original uncoded frame X.

In the derivation of the distortion between $\hat{X}$ and $\tilde{X}$, we summarize the features of conventional motion-compensated predictive video codecs as those of the MPEG-x and H.26x families. According to these standards, each frame of a video sequence is partitioned into nonoverlapping regions of $B \times B = 16 \times 16$ pixels called MBs. Each macroblock can be coded exploiting either the spatial redundancy (intra-macroblock coding) or the temporal redundancy (inter-macroblock coding). Coded data relative to macroblocks are gathered into slices, then packetized and transmitted through a noisy channel, that drops packets according to a given PLR. At the receiver side, macroblocks belonging to lost packets cannot be decoded. Therefore the decoder tries to partially recover lost data by means of an error concealment algorithm. Lost data cannot be perfectly recovered and channel distortion is inevitably introduced. Moreover, the inter-macroblock coding allows channel errors occurred in previously decoded frames to propagate along the decoded video sequence, affecting also those macroblocks for which data have been correctly received, thus introducing temporal error propagation.

Furthermore, the state-of-art H.264/AVC video coding standard [9] includes new coding tools that introduce spatial distortion propagation: intra-prediction, to efficiently exploit the spatial redundancy in smooth areas, and in-loop deblocking filter, to attenuate blocking artifacts due to block-based motion compensation and transform coding [22].

B. Derivation of the Channel Induced Distortion

The proposed algorithm aims at estimating the channel induced distortion at the macroblock granularity according to the mean square error (MSE) metrics:

$$D_n^i = \frac{1}{B^2}\sum_{x=1}^{B}\sum_{y=1}^{B}\left(\mathbf{E}_n^i(x,y)\right)^2 = \frac{1}{B^2}\sum_{x=1}^{B}\sum_{y=1}^{B}\left(\hat{\mathbf{M}}_n^i(x,y) - \tilde{\mathbf{M}}_n^i(x,y)\right)^2 \quad i = 1,\ldots,N \quad (1)$$

where N is the number of macroblocks, $\hat{\mathbf{M}}_n^i$ denotes a $B \times B$ matrix representing the ith macroblock in frame n reconstructed at the encoder side (i.e., available at the decoder in the error free scenario), and $\tilde{\mathbf{M}}_n^i$ the same macroblock reconstructed at the decoder side when channel losses occurred. Since the MSE is an additive metric, it can be readily computed at frame or sequence level by summing up the contributions of the individual macroblocks.

In the derivation of an estimator $\hat{D}_n^i$ for the quantity in (1), the following notation will be used (for the ith macroblock in frame n):

- $\mathbf{P}_n^i$: a $B \times B$ matrix representing the spatial or temporal predictor. More specifically, let $\hat{\mathbf{P}}_n^i$ and $\tilde{\mathbf{P}}_n^i$ denote, respectively, the predictor available at the encoder and decoder side;
- $\mathbf{\Theta}_n^i$: a $B \times B$ matrix representing prediction residuals;
- $\mathbf{v}_n^i = (v_{n,x}^i, v_{n,y}^i)$: motion vector with its scalar components along, respectively, the horizontal and vertical dimensions;
- $\tilde{\mathbf{v}}_n^i = (\tilde{v}_{n,x}^i, \tilde{v}_{n,y}^i)$: motion vector computed by the concealment algorithm with its components along, respectively, the horizontal and vertical dimensions;
- $\mathbf{E}_n^i = \hat{\mathbf{M}}_n^i - \tilde{\mathbf{M}}_n^i$: a $B \times B$ matrix representing channel induced errors.

The channel induced distortion is modeled distinguishing between whether a macroblock has been correctly received or not and taking into account the predictive (both spatial and temporal) nature of the H.264/AVC codec, together with the concealment algorithm adopted by the decoder. In fact, when a macroblock is correctly received, the decoder can reconstruct its pixel values, although errors might propagate from frames used as reference in the motion-compensation phase (temporal error propagation) or from neighboring reconstructed pixel values in the same frame (spatial error propagation). Conversely, when a macroblock is lost, errors are inevitably introduced in the areas corresponding to the missing pixels. Furthermore, depending on the decoder concealment strategy, errors can propagate either along the temporal or spatial dimension. Therefore, we can envisage the following scenarios:

1) *Correctly received intra-predicted macroblock*—intra_ok: the reconstructed pixel values of the macroblock at the decoder are obtained adding the predictor to the residuals, $\tilde{\mathbf{M}}_n^i = \mathbf{P}_n^i + \mathbf{\Theta}_n^i$. The channel induced error is given by

$$\begin{aligned}\mathbf{E}_n^i\{\text{intra\_ok}\} &= \hat{\mathbf{M}}_n^i - \tilde{\mathbf{M}}_n^i \\ &= \left(\hat{\mathbf{P}}_n^i + \mathbf{\Theta}_n^i\right) - \left(\tilde{\mathbf{P}}_n^i + \mathbf{\Theta}_n^i\right) \\ &= \left(\hat{\mathbf{P}}_n^i - \tilde{\mathbf{P}}_n^i\right) = \mathbf{E}_n^{i,\mathrm{SP}}. \end{aligned} \quad (2)$$

Equation (2) accounts also for spatial propagation (SP) of channel errors. Indeed, the spatial predictors $\hat{\mathbf{P}}_n^i$ and $\tilde{\mathbf{P}}_n^i$ might differ, since they are formed from (possibly different) previously reconstructed pixel values, according to the intra-prediction modes allowed by the H.264/AVC standard [9]. The computation of the spatial predictors $\mathbf{P}_n^i$ can be formalized as follows:

$$\hat{\mathbf{P}}_n^i(x,y) = \sum_{l=1}^{L_{x,y}} \alpha(l) \cdot \hat{X}_n(j(l;x,y), k(l;x,y)) \quad (3)$$

$$\tilde{\mathbf{P}}_n^i(x,y) = \sum_{l=1}^{L_{x,y}} \alpha(l) \cdot \tilde{X}_n(j(l;x,y), k(l;x,y)) \quad (4)$$

where $L_{x,y}$ represents the cardinality of the set of previously reconstructed pixel values used to interpolate pixel (x,y) and $(j(l;x,y),k(l;x,y)), l = 1,\ldots,L_{x,y}$, are the corresponding spatial locations. Finally $\alpha(l)$ are the weighted coefficients whose values are defined by the standard depending on the intra-prediction mode and satisfy the constraint $\sum_{l=1}^{L_{x,y}} \alpha(l) = 1$.

2) *Correctly received inter predicted macroblock*—inter_ok: as in the previous case the reconstructed pixel values are given by $\tilde{\mathbf{M}}_n^i = \mathbf{P}_n^i + \mathbf{\Theta}_n^i$ and the channel induced error is given by

$$\begin{aligned}\mathbf{E}_n^i\{\text{inter\_ok}\} &= \left(\hat{\mathbf{P}}_n^i + \mathbf{\Theta}_n^i\right) - \left(\tilde{\mathbf{P}}_n^i + \mathbf{\Theta}_n^i\right) \\ &= \left(\hat{\mathbf{P}}_n^i - \tilde{\mathbf{P}}_n^i\right) = \mathbf{E}_n^{i,\text{TP}}. \end{aligned} \tag{5}$$

The temporal propagation (TP) of channel errors is due to a mismatch in the pixel values used by the motion-compensation to generate $\mathbf{P}_n^i$:

$$\hat{\mathbf{P}}_n^i(x,y) = \hat{X}_{n-r}(x^i + x + v_{n,x}^i, y^i + y + v_{n,y}^i) \tag{6}$$

$$\tilde{\mathbf{P}}_n^i(x,y) = \hat{X}_{n-r}(x^i + x + v_{n,x}^i, y^i + y + v_{n,y}^i) \tag{7}$$

where (x^i, y^i) are the coordinates of the top-left pixel of the ith macroblock and r denotes the index of the reference frame used for motion compensation to account for the multiple reference frame (MRF) coding tool adopted in the H.264/AVC standard [9]. The reference frame is spatially interpolated if the motion vector $\mathbf{v}_n^i$ has sub-pixel accuracy.

3) *Lost macroblock and spatial concealment*—intra_ko: the decoder computes a concealed macroblock $\tilde{\mathbf{M}}_n^i$ by means of spatial interpolation with the pixels of surrounding macroblocks $\tilde{\mathbf{M}}_n^i$:

$$\tilde{\mathbf{M}}_n^i(x,y) = \sum_{l=1}^{L_{x,y}} \beta(l) \cdot \tilde{X}_n(j(l;x,y), k(l;x,y)) \tag{8}$$

where $L_{x,y}$ is the cardinality of the set of pixels involved in the spatial concealment of pixel (x,y), and $(j(l;x,y),k(l;x,y)), l = 1,\ldots,L_{x,y}$, are the locations of the pixels involved in the interpolation. Finally, $\beta(l)$ are the weighted coefficients used by the concealment algorithm such that $\sum_{l=1}^{L_{x,y}} \beta(l) = 1$. The channel induced error $\mathbf{E}_n^i$ is given by

$$\begin{aligned}\mathbf{E}_n^i\{\text{intra\_ko}\} &= \hat{\mathbf{M}}_n^i - \tilde{\mathbf{M}}_n^i \\ &= \hat{\mathbf{M}}_n^i - \bar{\mathbf{M}}_n^i + \bar{\mathbf{M}}_n^i - \tilde{\mathbf{M}}_n^i \\ &= \left(\hat{\mathbf{M}}_n^i - \bar{\mathbf{M}}_n^i\right) + \left(\bar{\mathbf{M}}_n^i - \tilde{\mathbf{M}}_n^i\right) \\ &= \mathbf{E}_n^{i,\text{SC}} + \bar{\mathbf{E}}_n^{i,\text{SP}} \end{aligned} \tag{9}$$

where the term $\bar{\mathbf{M}}_n^i$ represents the concealed macroblock obtained as in (8) by replacing $\hat{X}_n$ with $\hat{X}_n$. Equation (9) highlights two contributions to the channel induced error $\mathbf{E}_n^i\{\text{intra\_ko}\}$: $\mathbf{E}_n^{i,\text{SC}}$ and $\bar{\mathbf{E}}_n^{i,\text{SP}}$. The presence of the first term is due to the fact that the spatial concealment (SC) is unable to perfectly reconstruct the texture of the missing macroblock. Since it represents the new error introduced at time n, it as an innovation term. Conversely, the second one represents a propagation term, specifically, the spatial propagation (SP) from previously reconstructed pixels in the same frame.

4) *Lost macroblock and temporal concealment*—inter_ko: the temporal concealment strategy replaces the missing pixels with the ones pointed by the motion vector $\tilde{\mathbf{v}}_n^i$ in the reference frame indexed by r:

$$\tilde{\mathbf{M}}_n^i(x,y) = \hat{X}_{n-r}\left(x^i + x + \tilde{v}_{n,x}^i, y^i + y + \tilde{v}_{n,y}^i\right). \tag{10}$$

In our work, we use the temporal concealment algorithm included in the H.264/AVC reference software [23], [24]. This algorithm selects the motion vector $\tilde{\mathbf{v}}_n^i$ according to a boundary absolute difference (BAD) distortion metric, which measures the difference between values of the concealed pixels and the reconstructed pixels surrounding the current macroblock [23]. The channel induced error is given by

$$\begin{aligned}\mathbf{E}_n^i\{\text{inter\_ko}\} &= \hat{\mathbf{M}}_n^i - \tilde{\mathbf{M}}_n^i \\ &= \hat{\mathbf{M}}_n^i - \bar{\mathbf{M}}_n^i + \bar{\mathbf{M}}_n^i - \tilde{\mathbf{M}}_n^i \\ &= \left(\hat{\mathbf{M}}_n^i - \bar{\mathbf{M}}_n^i\right) + \left(\bar{\mathbf{M}}_n^i - \tilde{\mathbf{M}}_n^i\right) \\ &= \mathbf{E}_n^{i,\text{TC}} + \bar{\mathbf{E}}_n^{i,\text{TP}} \end{aligned} \tag{11}$$

where the term $\bar{\mathbf{M}}_n^i$ denotes the concealed macroblock obtained as in (10) by replacing $\tilde{X}_{n-r}$ with $\hat{X}_{n-r}$. As in (9), also (11) highlights two contributions. The first term takes into account the fact that the temporal concealment algorithm is unable to perfectly reconstruct the pixel values of the lost macroblock (TC—innovation term), while the second term considers temporal error propagation (TP—propagation term).

The contribution to the distortion due to the innovation $(\mathbf{E}_n^{i,\text{SC}}, \mathbf{E}_n^{i,\text{TC}})$ and propagation $(\mathbf{E}_n^{i,\text{SP}}, \mathbf{E}_n^{i,\text{TP}})$ terms will be derived in the following. We notice that the choice of the concealment strategy (spatial or temporal) when a macroblock is lost depends on the decoder implementation, the frame type the lost macroblock belongs to and also by the adopted distortion measure [24]. Hereafter, we will follow the convention adopted in the decoder reference software model [25], [23]: macroblocks belonging to intra-coded frames will be spatially concealed, whilst macroblocks belonging to inter coded frames (P or B-slice) will be temporally concealed.

We also notice that the channel induced errors derived in (2), (5), (9), and (11) do not account for the effect produced by the deblocking filter. This coding tool is a spatial filter that smooths the artifacts introduced by both block based motion compensation and transform coding, while preserving the real image edges. The processing performed by the deblocking filter, causes the channel induced errors to propagate along the spatial dimension also affecting correctly received macroblocks, which were not intra-predicted and where the temporal error propagation did not occur. The deblocking filter contribution to the final channel induced error (and also the distortion) will be discussed in Section III-G.

To compute the mean square error distortion, we substitute (2), (5), (9), and (11) into (1) to obtain

$$\begin{aligned} D_n^i\{\text{intra\_ok}\} &= \frac{1}{B^2}\sum_{x=1}^{B}\sum_{y=1}^{B}\left(\mathbf{E}_n^{i,\text{SP}}(x,y)\right)^2 \\ &= D_n^{i,\text{SP}} \end{aligned} \tag{12}$$

$$\begin{aligned} D_n^i\{\text{inter\_ok}\} &= \frac{1}{B^2}\sum_{x=1}^{B}\sum_{y=1}^{B}\left(\mathbf{E}_n^{i,\text{TP}}(x,y)\right)^2 \\ &= D_n^{i,\text{TP}} \end{aligned} \tag{13}$$

$$
\begin{aligned}
D_n^i\{\text{intra\_}ko\} &= \frac{1}{B^2}\sum_{x=1}^{B}\sum_{y=1}^{B}\big(\mathbf{E}_n^{i,\mathrm{SC}}(x,y) \\
&\quad +\mathbf{E}_n^{i,\mathrm{SP}}(x,y)\big)^2 \\
&= \frac{1}{B^2}\sum_{x=1}^{B}\sum_{y=1}^{B}\big(\mathbf{E}_n^{i,\mathrm{SC}}(x,y)\big)^2 \\
&\quad + \frac{1}{B^2}\sum_{x=1}^{B}\sum_{y=1}^{B}\big(\mathbf{E}_n^{i,\mathrm{SP}}(x,y)\big)^2 \\
&\quad + \frac{2}{B^2}\sum_{x=1}^{B}\sum_{y=1}^{B}\big(\mathbf{E}_n^{i,\mathrm{SC}}(x,y) \\
&\quad \cdot\mathbf{E}_n^{i,\mathrm{SP}}(x,y)\big) \\
&= D_n^{i,\mathrm{SC}} + D_n^{i,\mathrm{SP}} + D_n^{i,\mathrm{SCP}} \qquad (14)
\end{aligned}
$$

$$
\begin{aligned}
D_n^i\{\text{inter\_ko}\} &= \frac{1}{B^2}\sum_{x=1}^{B}\sum_{y=1}^{B}\big(\mathbf{E}_n^{i,\mathrm{TC}}(x,y) \\
&\quad +\mathbf{E}_n^{i,\mathrm{TP}}(x,y)\big)^2 \\
&= D_n^{i,\mathrm{TC}} + D_n^{i,\mathrm{TP}} + D_n^{i,\mathrm{TCP}}. \qquad (15)
\end{aligned}
$$

The last terms in the right hand sides of (14) and (15) represent the correlation between the innovation and the propagation terms. As far as temporal error propagation is concerned, the work in [26] states that the term $D_n^{i,\mathrm{TCP}}$ is not equal to zero, especially when burst errors occur during transmission. Nevertheless, we have experimentally measured the term $D_n^{i,\mathrm{TCP}}$ and found that it is almost negligible with respect to the overall distortion $D_n^i\{\text{inter\_ko}\}$, as shown in Table I(a) and (b). These results are obtained by applying error patterns with PLR in the range [0.1%, 20%] generated with a two state Gilbert's model [27], under the same test conditions further detailed in Section V, averaging over all the tested video sequences. The work in [28], considers a maximum burst length of nine packets as characteristic of IP networks, with an average around two to three packets. In our experiments we tuned the model parameters to obtain an average burst length of, respectively, three and eleven packets, where each packet consists of a row of macroblocks. Similar results are also obtained for the spatial term $D_n^{i,\mathrm{SCP}}$. Therefore, in the following, we will set both $\hat{D}_n^{i,\mathrm{TCP}}$ and $\hat{D}_n^{i,\mathrm{SCP}}$ terms equal to zero.

In order to summarize previous derivations, the pseudo code in Algorithm 1 represents the execution steps of the proposed NORM algorithm. In the following sections we show how the proposed algorithm computes the estimates $\hat{D}_n^{i,\mathrm{SP}}, \hat{D}_n^{i,\mathrm{TP}}, \hat{D}_n^{i,\mathrm{SC}}$ and $\hat{D}_n^{i,\mathrm{TC}}$ of the quantities appearing in the expressions (12)–(15).

Algorithm 1: Pseudocode for the proposed NORM algorithm

1) **for** $n = 0$ to $N-1$ **do**
2) **for** $i = 0$ to $M-1$ **do**
3) **if** macroblock i is lost **then**
4) **if** macroblock $i \in$ I frame **then**
5) $\hat{D}_n^i = \hat{D}_n^i\{\text{intra\_ko}\} = \hat{D}_n^{i,\mathrm{SC}} + \hat{D}_n^{i,\mathrm{SP}}$ {see Section III.E and III.C}
6) **else**
7) $\hat{D}_n^i = \hat{D}_n^i\{\text{inter\_ko}\} = \hat{D}_n^{i,\mathrm{TC}} + \hat{D}_n^{i,\mathrm{TP}}$ {see Section III.F and III.D}
8) **end if**
9) **else**
10) **if** macroblock $i \in$ P or B frame **then**
11) $\hat{D}_n^i = \hat{D}_n^i\{\text{intra\_ok}\} = \hat{D}_n^{i,\mathrm{SP}}$ {see Section III.C}
12) **else**
13) $\hat{D}_n^i = \hat{D}_n^i\{\text{inter\_ok}\} = \hat{D}_n^{i,\mathrm{TP}}$ {see Section III.D}
14) **end if**
15) **end if**
16) **end for**
17) **end for**

TABLE I
AVERAGE MEASURED VALUES OF THE DISTORTION TERMS $D^{\mathrm{TCP}}, D\{\text{inter\_ko}\}, D^{\mathrm{SCP}}$, AND $D\{\text{intra\_ko}\}$ WHEN THE AVERAGE BURST LENGTH IS EQUAL TO THREE AND ELEVEN. THE AVERAGE IS COMPUTED OVER LOST MACROBLOCKS ONLY. (A) AVERAGE BURST $\mathrm{length} = 3$. (B) AVERAGE BURST $\mathrm{length} = 11$

(a)

PLR [%]	D^{TCP}	$D\{inter\_ko\}$	D^{SCP}	$D\{intra\_ko\}$
0.1	0.550	19.40	0.504	46.40
0.4	0.615	26.25	0.252	50.38
0.7	1.333	34.85	1.191	66.95
1.0	1.730	33.34	0.100	77.91
1.3	1.650	41.23	0.582	81.61
1.6	2.000	50.47	1.626	94.65
1.9	2.217	65.64	0.022	108.53
2.2	2.473	83.36	0.343	123.53
2.5	2.07	87.55	0.042	129.11
3	1.72	167.09	2.94	274.25
5	2.86	251.70	4.71	469.88
10	3.04	292.06	7.07	604.24
20	9.92	549.86	17.66	1332.13

(b)

PLR [%]	D^{TCP}	$D\{inter\_ko\}$	D^{SCP}	$D\{intra\_ko\}$
0.1	6.66	190.08	1.54	370.69
0.4	6.97	553.53	1.79	389.44
0.7	7.21	590.76	3.44	620.31
1.0	8.63	602.30	6.63	693.49
1.3	9.61	691.06	6.78	769.57
1.6	10.73	661.73	8.53	722.59
1.9	23.63	872.85	10.63	956.41
2.2	25.45	896.78	13.61	990.67
2.5	29.71	926.60	20.78	1093.87
3	38.33	916.63	84.95	2035.15
5	40.61	1282.43	87.67	2598.30
10	45.23	1896.20	94.84	3461.35
20	50.23	2059.17	105.13	3752.35

C. Spatial Error Propagation

The H.264/AVC video coding standard allows intra macroblock prediction at two different levels of granularity: 16×16 pixels macroblock or 4×4 pixels sub-block. For each case there are, respectively, four and nine different modes to compute the spatial predictor [9]. Therefore, the channel induced errors can spatially propagate from each pixel involved in the spatial predictor computation.

The work in [29] accurately models the distortion due to the spatial propagation starting from the pixel level granularity. Conversely, the proposed NORM algorithm works at the macroblock level granularity. We have found in our experiments that the distortion term $D_n^{i,\mathrm{SP}}$ accounts for a negligible fraction of the overall macroblock distortion D_n^i in intra-frames as shown in Table II. The measurements have been averaged over the test video sequences and 30 channel realizations with PLR

TABLE II
MEASURED VALUES OF D AND D^{SP} COMPUTED OVER ALL MACROBLOCKS BELONGING TO INTRA-FRAMES

PLR [%]	D	D^{SP}
0.1	17.92	0.085
0.4	20.77	0.096
0.7	21.08	0.108
1.0	27.72	0.099
1.3	32.66	0.128
1.6	34.90	0.114
1.9	36.29	0.157
2.2	37.69	0.193
2.5	39.47	0.208
3	40.75	0.173
5	45.33	0.184
10	56.61	0.703
20	135.62	1.345

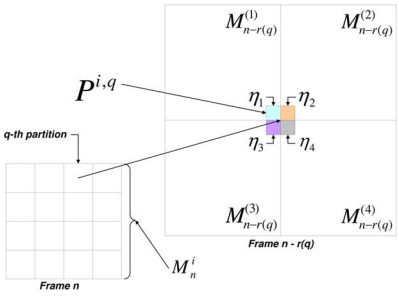

Fig. 2. Temporal distortion propagation when the predictor for the qth motion vector $\mathbf{P}^{i,q}$ overlaps with four macroblocks in the reference frame $n - r(q)$.

in the range [0.1%, 20%]. Therefore, in our work we neglect the contribution of the spatial propagation term and we set

$$\hat{D}_n^{i,\mathrm{SP}} = 0. \tag{16}$$

D. Temporal Error Propagation

Channel losses occurred in previous frames forward propagate due to the motion-compensation loop performed at the decoder. Each macroblock can be partitioned into up to sixteen sub-blocks, each with a motion vector assigned to it. Let $\mathbf{v}_n^{i,q}$ denote the motion vector related to the qth partition in macroblock i. To simplify the notation, we consider that every macroblock is always split into 16 partitions of 4×4 sub-blocks, each with a corresponding motion vector. If the coding mode is such that the macroblock is split into a fewer number of partitions, the motion vectors are replicated for each 4×4 sub-blocks. The temporal predictor $\tilde{\mathbf{P}}_n^i$ of a macroblock is computed at the decoder combining the predictors of each 4×4 sub-block partition $\tilde{\mathbf{P}}_n^{i,q}, q = 1, \ldots, 16$. Fig. 2 illustrates that each predictor $\tilde{\mathbf{P}}_n^{i,q}$ overlaps with $N_0(q) = 4$ macroblocks (although in general, $1 \leq N_0(q) \leq 4$) at time $n - r(q)$, where the reference frame index r depends also on q, since the MRF coding tool allows a different reference frame for each sub-block.[1] In our algorithm, we estimate the distortion propagated at time n as a weighted average of the distortion contributions previously computed for those macroblocks the current sub-block overlaps with.

[1]The H.264/AVC standard restricts the 4×4 pixels partitions to have the same reference frame as the one of the 8×8 pixels partition where they belong to [30].

Let ζ_p denote the number of pixels in $\tilde{\mathbf{P}}^{i,q}$ that overlap with the pth macroblock ($1 \leq p \leq N_0(q)$). The estimate of the temporal propagation term is given by

$$\hat{D}_n^{i,\mathrm{TP}} = \frac{1}{16} \cdot \sum_{q=1}^{16} \left(\sum_{p=1}^{N_0(q)} \eta_p \cdot \hat{D}^{(p)}_{n-r(q)} \right) \quad \text{with} \quad \eta_p = \frac{\zeta_p}{16} \tag{17}$$

where the superscript $^{(p)}$ denotes the local indexing of the reference macroblocks induced by the predictor $\mathbf{P}^{i,q}$, as shown in Fig. 2.

The estimate given in (17), is used for temporal distortion propagation when either the macroblock is correctly received (to compute $\hat{D}_n^i\{\mathrm{inter\_ok}\}$) or it has been lost (to compute $\hat{D}_n^i\{\mathrm{inter\_ko}\}$). In this latter case we set $\mathbf{v}_n^{i,q} = \tilde{\mathbf{v}}_n^i, \forall q$.

E. Error Introduced by Spatial Concealment

The distortion due to the action of the spatial concealment $D_n^{i,\mathrm{SC}}$ is related to the loss of high frequency content of the lost macroblock. In fact, the spatial concealment algorithm implemented in the H.264/AVC reference software performs a simple spatial interpolation using pixel values of neighboring reconstructed macroblocks. In our work, we estimate this contribution by comparing the spatially concealed macroblock $\tilde{\mathbf{M}}_n^i$ [see (8)] with the one obtained with a simple zero-motion temporal concealment $\tilde{\mathbf{M}}_n^{i,0}$ [see (10), with $(\tilde{v}_x^i, \tilde{v}_y^i) = (0, 0)$], where the latter typically preserves the high frequency content of the original block

$$\hat{D}_n^{i,\mathrm{SC}} = \frac{1}{B^2} \sum_{x=1}^{B} \sum_{y=1}^{B} \left(\tilde{\mathbf{M}}_n^i(x,y) - \tilde{\mathbf{M}}_n^{i,0}(x,y) \right)^2. \tag{18}$$

Our experimental findings show that the difference, in MSE sense, between the two concealment strategies in (18) is highly correlated (correlation coefficient in the range 0.85–0.97) with the actual MSE distortion between the spatially concealed and original macroblocks. Although, in some cases, a simple zero motion concealment strategy might lead to a lower distortion, in this work we do not alter the default concealment strategy implemented in the H.264/AVC reference software.

F. Error Introduced by Temporal Concealment

When a macroblock that belongs to a P or B frame is lost, both the coding modes, motion vectors (MV) and the prediction residuals (PR) cannot be recovered. In this work, we estimate the distortion due to temporal concealment $D_n^{i,\mathrm{TC}}$ as the sum of two contributions:

$$\hat{D}_n^{i,\mathrm{TC}} = \hat{D}_n^{i,\mathrm{MV}} + \hat{D}_n^{i,\mathrm{PR}} \tag{19}$$

which might be reasonably considered to be uncorrelated. We notice that (19) assumes that the original mode is inter-frame coding. In fact, the loss of the coding mode does not allow to determine if the macroblock has been originally intra-coded. Nevertheless, in our experiments we verified that intra-coded macroblocks in P and B slices did not exceed 4% of the total number of macroblocks, thus validating our assumption.

In the following, we show how the two terms in (19) are actually estimated:

1) *Distortion induced by the lack of motion vector*: In the derivation of the model to estimate $D_n^{i,\mathrm{MV}}$, we make the simplifying assumption that macroblock i has only one motion vector $\bar{\mathbf{v}}^i = (\bar{v}_x^i, \bar{v}_y^i)$ defined as the arithmetic mean of the original motion vectors of the constituent sub-blocks:

$$\bar{\mathbf{v}}_n^i = \frac{1}{16} \cdot \sum_{q=1}^{16} \mathbf{v}_n^{i,q}. \tag{20}$$

In case of translational motion, the difference between the predictor provided by the temporal concealment algorithm $\tilde{\mathbf{P}}_n^i$ and the one corresponding to the original motion vector estimated at the encoder, $\hat{\mathbf{P}}_n^i$, consists of a spatial shift related to the difference between the original and the concealed motion vectors, i.e., $\bar{\mathbf{v}}^i$ and $\tilde{\mathbf{v}}^i$, respectively [31]:

$$\tilde{\mathbf{P}}_n^i(x,y) = \hat{\mathbf{P}}_n^i\left(x - \delta_{n,x}^i, y - \delta_{n,y}^i\right) \tag{21}$$

where $\delta_{n,x}^i = \bar{v}_{n,x}^i - \tilde{v}_{n,x}^i$ and $\delta_{n,y}^i = \bar{v}_{n,y}^i - \tilde{v}_{n,y}^i$. By applying the shift theorem of the discrete time Fourier transform (DTFT) to (21) we obtain a phase rotation, i.e., $\tilde{\mathbf{P}}_n^i(\boldsymbol{\omega}) = \mathbf{P}_n^i(\boldsymbol{\omega}) \cdot e^{-j(\boldsymbol{\omega}\cdot\boldsymbol{\delta}_n^i)}$, with $\boldsymbol{\omega} = (\omega_x, \omega_y)$ and $\boldsymbol{\delta}_n^i = (\delta_{n,x}^i, \delta_{n,y}^i)$. According to the Parseval's theorem, the estimated distortion due to the lack of motion vectors is given by

$$\begin{aligned}\hat{D}_n^{i,\mathrm{MV}} &= \frac{1}{(2\pi)^2} \cdot \int_{-\pi}^{\pi}\int_{-\pi}^{\pi} \boldsymbol{\Phi}_n^i(\boldsymbol{\omega}) |1 - e^{-j\left(\boldsymbol{\omega}\cdot\boldsymbol{\delta}_n^i\right)}|^2 d\boldsymbol{\omega} \\ &= \frac{1}{(2\pi)^2} \cdot \int_{-\pi}^{\pi}\int_{-\pi}^{\pi} \boldsymbol{\Phi}_n^i(\boldsymbol{\omega}) \left(2 - 2\cos(\boldsymbol{\omega}\cdot\boldsymbol{\delta}_n^i)\right) d\boldsymbol{\omega}\end{aligned} \tag{22}$$

where the term $\boldsymbol{\Phi}_n^i(\boldsymbol{\omega})$ denotes the power spectral density of $\hat{\mathbf{P}}_n^i(\boldsymbol{\omega})$. Since the signals have finite support, we can sample the DTFT to obtain a discrete frequency version of (22):

$$\hat{D}_n^{i,\mathrm{MV}} \simeq \frac{1}{B^4} \sum_{j=0}^{B-1}\sum_{k=0}^{B-1} \boldsymbol{\Phi}_n^i(\omega_j, \omega_k) \cdot(1 - \cos(\omega_j\delta_x + \omega_k\delta_y)) \tag{23}$$

where the discrete frequencies ω_j, ω_k are equal to, respectively, $2\pi j/B$ and $2\pi k/B$. Strict equality in (23) would hold only if the predictors were periodic in both directions with period B outside the macroblock boundaries and the pure translatory motion hypothesis holds. The term $\boldsymbol{\Phi}(\omega_j, \omega_k), j = 1, \ldots, B, k = 1, \ldots, B$ can be computed by means of the 2D-DFT

$$\boldsymbol{\Phi}_n^i(\omega_j, \omega_k) = \left| \frac{1}{B^2} \sum_{x=0}^{B-1}\sum_{y=0}^{B-1} \hat{\mathbf{P}}_n^i(x,y) e^{-j(\omega_j x + \omega_k y)} \right|^2. \tag{24}$$

We notice from (23) that the estimation of $\hat{D}_n^{i,\mathrm{MV}}$ involves two unknown quantities. To obtain $\boldsymbol{\Phi}_n^i(\boldsymbol{\omega})$ we replace the (missing) original predictor $\hat{\mathbf{P}}_n^i$ with the one obtained with temporal concealment $\tilde{\mathbf{P}}_n^i$. On the other hand, the term $\boldsymbol{\delta}_n^i$ implies the knowledge of the original average motion vector $\bar{\mathbf{v}}_n^i$. We found that a reasonable estimate of $\boldsymbol{\delta}_n^i$ for the problem at hand is given by the following expression:

$$\hat{\boldsymbol{\delta}}_n^i = \sqrt{\frac{1}{L} \cdot \sum_{l=1}^{L_\varphi} \left(\tilde{\mathbf{v}}_n^i - \mathbf{v}_{8\times8}^l\right)^2} \tag{25}$$

where L_φ is the cardinality of the set of candidate motion vectors related to 8×8 neighboring sub-blocks used by the temporal concealment algorithm to compute the concealed motion vector $\tilde{\mathbf{v}}_n^i$ (refer to Fig. 3). The rationale is that the concealed motion vector likely differs from the original one if we observe a large variability among candidate motion vectors. In order to quantify the accuracy of the estimate $\hat{\boldsymbol{\delta}}_n^i$ we measured the Pearson's correlation coefficient between $|\boldsymbol{\delta}_n^i|$ and $\hat{\boldsymbol{\delta}}_n^i$ for the same video sequences and test conditions adopted for our experiments in Section V. We are interested only in the absolute value of the displacement error, since the distortion does not depends on its sign. We notice from the results in Table III that a good correlation (higher than 0.85) exists.

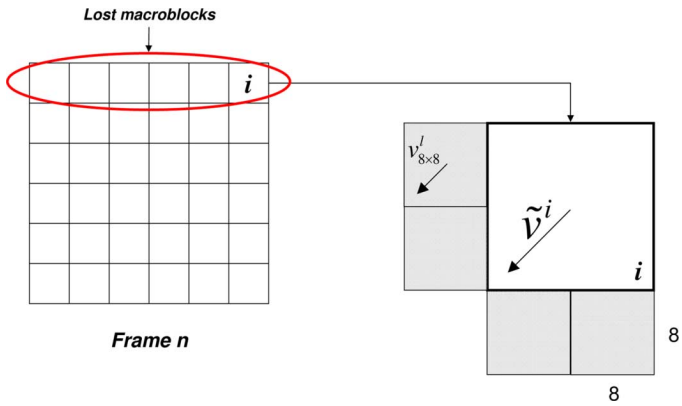

Fig. 3. Motion vectors $\mathbf{v}_{8\times8}^l$ used in the estimate of $\boldsymbol{\delta}$. Grey 8×8 blocks belong to the L_φ set.

TABLE III
PEARSON'S CORRELATION COEFFICIENT BETWEEN $|\boldsymbol{\delta}|$ AND $\hat{\boldsymbol{\delta}}$

PLR [%]	$\rho(\lvert\delta_x\rvert, \hat{\delta}_x)$	$\rho(\lvert\delta_y\rvert, \hat{\delta}_y)$
0.1	0.92	0.90
0.4	0.90	0.90
0.7	0.89	0.87
1.0	0.90	0.86
1.3	0.88	0.89
1.6	0.89	0.88
1.9	0.92	0.89
2.2	0.91	0.86
2.5	0.92	0.88
3	0.92	0.86
5	0.89	0.87
10	0.92	0.93
20	0.92	0.92

2) *Distortion induced by the lack of prediction residuals*: There is no way of retrieving the prediction residuals when a macroblock is lost, since they are simply set to zero. Nevertheless, we observe that the energy of prediction residuals is higher in regions where occlusions occur. Since there is temporal correlation between occlusions in

TABLE IV
AVERAGE MEASURED VALUES OF THE DISTORTION INDUCED BY CHANNEL LOSSES AND SPATIAL PROPAGATION DUE TO THE DEBLOCKING FILTERING

PLR [%]	D	$D\{deblocking\}$
0.1	3.89	0.036
0.4	6.13	0.048
0.7	5.48	0.071
1.0	6.23	0.066
1.3	8.65	0.096
1.6	8.61	0.102
1.9	9.06	0.095
2.2	11.57	0.215
2.5	11.86	0.246
3	10.75	0.171
5	25.33	0.482
10	46.61	0.703
20	114.62	1.129

neighboring frames, we decide to estimate $\hat{D}_n^{i,\mathrm{PR}}$ with the energy of the residuals in the reference frame pointed by the concealed motion vector.

$$\hat{D}_n^{i,\mathrm{PR}} = \frac{1}{B^2}\sum_{x=1}^{B}\sum_{y=1}^{B}\Theta_{n-r}^{i}\left(x+\tilde{v}_x^i, y+\tilde{v}_y^i\right)^2. \qquad (26)$$

G. Deblocking Filter Error Propagation

The H.264/AVC video coding standard adopts an in-loop spatial filter to smooth blocking artifacts introduced by block-based motion compensation and transform coding. The adaptive deblocking filter [22] performs spatial interpolation between pixels at the boundaries of 4×4 sub-blocks; it provides better rate-distortion performance and enhances the final decoded image quality [9]. The overall effect of the deblocking filter is to spatially propagate the channel induced distortion and, at the same time, to attenuate the temporal propagation of the distortion due to spatial filtering [2]. The work in [29] explicitly models the effect of deblocking filter at the pixel level. Conversely, in our work we are interested in describing the effect at the macroblock level. Indeed, in our experiments we found that at the macroblock level the distortion propagation due to deblocking filter is rather limited, as shown in Table IV.

H. Computational Complexity

We provide an approximate indication of the computational cost required by NORM, showing that it grows linearly with the frame size. The exact cost obviously depends on the specific hardware/software implementation. For each macroblock, the NORM algorithm computes the contribution to the distortion due to temporal error propagation $D_n^{i,\mathrm{TP}}$ by means of (17). For each of the 4×4 sub-blocks, we need to compute the weights η_p. This requires 2 integer arithmetic divisions (to obtain the coordinates of the macroblock $\mathbf{M}^{(1)}_{n-r(q)}$ in Fig. 2) and 4 subtractions/shifts. Then up to 4 floating point multiplications/additions are needed to compute the inner sum in (17). An upper bound of the cost of is of the order of $16(2+4+4+4)/B^2 = 14/16$ operations per pixel. When a macroblock is lost, we need to compute the extra terms to account for the new errors introduced. We consider here the case of a macroblock lost in a inter-frame coded slice, i.e., $D_n^{i,\mathrm{TC}}$. To compute the distortion due to the loss of prediction residuals $D_n^{i,\mathrm{PR}}$ as in (26) requires B^2 multiplications and additions. As for the distortion introduced by the loss of motion vectors $D_n^{i,MV}$ in (23) we need $cB^2\log_2 B$ operations to compute the 2-D FFT (the factor c depends on the specific implementation of the FFT), B^2 operations for the element-wise product with $(1-\cos(\omega_j\delta_x+\omega_k\delta_y))$ [note that this term can be pre-computed for different values of (δ_x,δ_y)] and B^2 additions to compute the double summation in (23). The cost is $\mathrm{PLR}\cdot(B^2+B^2+cB^2\log_2 B+B^2+B^2)/B^2 = \mathrm{PLR}\cdot(4+4c)$ operations per pixel, i.e., grows linearly with the PLR. Therefore, for low PLR, most of the complexity is related to the computation of temporal error propagation.

IV. REDUCED REFERENCE QUALITY ASSESSMENT

Mean square error metrics have been shown to be only partially correlated to the visual quality of video sequences. In fact, it is easy to construct examples where two realizations of the same video sequence, characterized by different degradations but with the same average MSE, have significantly different perceived visual quality [8].

The NORM algorithm produces a no-reference estimate of the channel induced distortion $\hat{D}_n^i$ measured in terms of the MSE at the macroblock level. Given the fine granularity level at which the distortion is computed, we are able to successfully exploit this information to produce perceptually related visual quality metrics. As a simple example, the macroblock level distortion could be weighted at the frame level by means of perceptual foveation perceptual models, which capture the spatially-varying characteristics of human visual attention [32]. In this paper we explore another direction, whereby the estimated MSE distortion at the macroblock level is used together with some additional information about the original noiseless sequence to produce a perceptual visual quality assessment metrics in reduced-reference mode. To this end, we use as a benchmark the SSIM index (structural similarity metrics) [12], an objective metrics computed in full-reference mode, which shows a good correlation with MOS scores collected in subjective tests. The SSIM metrics has been first applied to images [12] and later extended to video sequences in [33] and [34]. The SSIM does not target a specific kind of distortion, rather it considers image degradations as perceived changes in structural information variation. The metrics is the result of three contributions: luminance, contrast and structural comparison between the original and the degraded image.

The goal of our work is to exploit the no-reference distortion estimate to reliably compute the SSIM index in reduced-reference mode, such as to limit the amount of extra information to be sent about the noiseless sequence.

A. Overall Setup

Fig. 4 depicts the overall setup for the proposed reduced-reference objective quality monitoring system. The nth frame of the original video sequence X is encoded by the H.264/AVC encoder. Once the reference frame $\hat{X}_n$ is available in the encoder reference frame buffer, the feature extraction module computes the mean value $\mu_{\hat{\mathbf{M}}_n^i}$ and the standard deviation $\sigma_{\hat{\mathbf{M}}_n^i}$ of each macroblock $\hat{\mathbf{M}}_n^i$. The encoded bitstream is then transmitted through the noisy channel that drops packets according to a given PLR. The quantities $\mu_{\hat{\mathbf{M}}_n^i}$ and $\sigma_{\hat{\mathbf{M}}_n^i}$ are also lossy encoded to obtain $\hat{\mu}_{\hat{\mathbf{M}}_n^i}$ and $\hat{\sigma}_{\hat{\mathbf{M}}_n^i}$ (details will be given in Section V) and transmitted through an error free channel as typically assumed

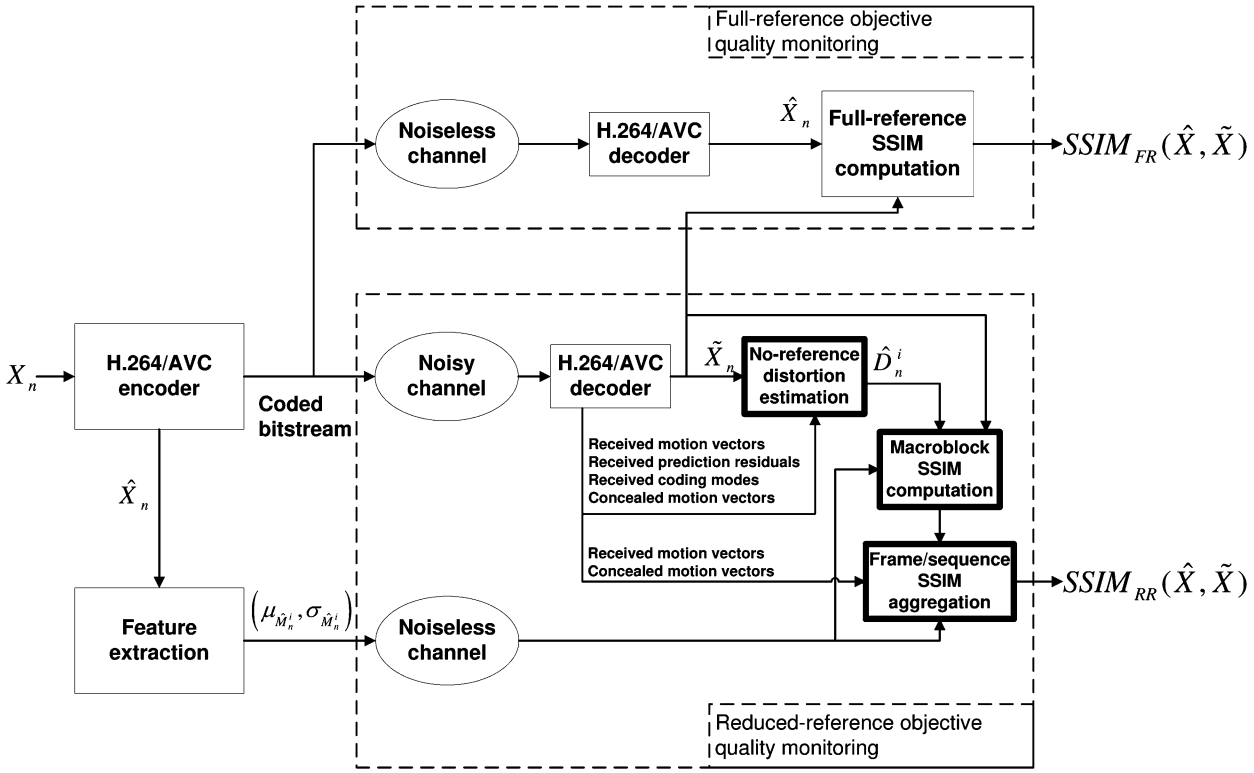

Fig. 4. Block diagram of the proposed reduced-reference system, which exploits the output of NORM to compute the SSIM score.

in the literature addressing reduced-reference systems [35]. At the decoder side, the channel induced distortion is estimated by the proposed algorithm described in Section III. The quantity $\hat{D}_n^i$ is used together with the reduced-reference information $\hat{\mu}_{\hat{\mathbf{M}}_n^i}$ and $\hat{\sigma}_{\hat{\mathbf{M}}_n^i}$ to compute $\mathrm{SSIM_{RR}}(\hat{\mathbf{M}}_n^i, \tilde{\mathbf{M}}_n^i)$. To obtain the SSIM metrics at the frame and sequence level, the SSIM values computed at the macroblock level are weighted in order to account for brightness and temporal masking, according to the algorithm described in [34]. To evaluate the accuracy of the proposed reduced-reference algorithm, we compare the $\mathrm{SSIM_{RR}}$ scores at the sequence level with the $\mathrm{SSIM_{FR}}$ scores computed in full-reference mode.

B. Reduced-Reference SSIM Computation

The full-reference SSIM metrics between the error-free macroblock $\hat{\mathbf{M}}_n^i$ and the concealed one $\tilde{\mathbf{M}}_n^i$ is given by the following expression [12], [33]:

$$\mathrm{SSIM_{FR}}(\hat{\mathbf{M}}_n^i, \tilde{\mathbf{M}}_n^i) = \frac{\left(2\mu_{\hat{\mathbf{M}}_n^i}\mu_{\tilde{\mathbf{M}}_n^i} + C_1\right)\cdot\left(2\sigma_{\hat{\mathbf{M}}_n^i,\tilde{\mathbf{M}}_n^i} + C_2\right)}{\left(\mu_{\hat{\mathbf{M}}_n^i}^2 + \mu_{\tilde{\mathbf{M}}_n^i}^2 + C_1\right)\cdot\left(\sigma_{\hat{\mathbf{M}}_n^i}^2 + \sigma_{\tilde{\mathbf{M}}_n^i}^2 + C_2\right)} \quad (27)$$

where $\sigma_{\hat{\mathbf{M}}_n^i,\tilde{\mathbf{M}}_n^i}$ represents the sample covariance between $\hat{\mathbf{M}}_n^i$ and $\tilde{\mathbf{M}}_n^i$:

$$\sigma_{\hat{\mathbf{M}}_n^i,\tilde{\mathbf{M}}_n^i} = \frac{1}{B^2}\sum_{x=1}^{B}\sum_{y=1}^{B}\left(\hat{\mathbf{M}}_n^i(x,y) - \mu_{\hat{\mathbf{M}}_n^i}\right) \times \left(\tilde{\mathbf{M}}_n^i(x,y) - \mu_{\tilde{\mathbf{M}}_n^i}\right). \quad (28)$$

The two constants C_1 ans C_2 are set as in [12] to avoid division by zero. To get the reduced-reference index, we replace $\mu_{\hat{\mathbf{M}}_n^i}$ and $\sigma_{\hat{\mathbf{M}}_n^i}$ with $\hat{\mu}_{\hat{\mathbf{M}}_n^i}$ and $\hat{\sigma}_{\hat{\mathbf{M}}_n^i}$, and with little algebra we obtain an approximation of the sample covariance by means of the available estimate $\hat{D}_n^i$ in (29) at the bottom of the page. The macroblock-level reduced-reference SSIM index is given by (30) at the bottom of the page. The SSIM at the frame and sequence-level is computed by a weighted average of the macroblock-level SSIM as detailed in [33].

V. Experimental Results and Comparisons

In order to validate both the proposed no-reference and reduced-reference systems, we carried out several experiments on real video sequences and a simulated error prone channel. In this section we discuss our results comparing them with other methods proposed in the literature.

$$\hat{\sigma}_{\hat{\mathbf{M}}_n^i,\tilde{\mathbf{M}}_n^i} = \frac{\left(\hat{\mu}_{\hat{\mathbf{M}}_n^i}^2 + \mu_{\tilde{\mathbf{M}}_n^i}^2 + \hat{\sigma}_{\hat{\mathbf{M}}_n^i}^2 + \sigma_{\tilde{\mathbf{M}}_n^i}^2\right) - \left(\hat{D}_n^i + 2\hat{\mu}_{\hat{\mathbf{M}}_n^i}^2\mu_{\tilde{\mathbf{M}}_n^i}^2\right)}{2} \quad (29)$$

TABLE V
H.264/AVC ENCODER PARAMETERS

Parameter	Value
Number of reference frames	5
Macroblock partitions for motion estimation	Enabled
Entropy coding	CAVLC, CABAC (for main profile)
Rate-distortion optimization	High complexity mode
Early skip detection mode decision	Enabled
Motion estimation algorithm	Enhanced predictive zonal search (EPZS)

A. Source Coding Conditions and Packetization Issues

We consider two target scenarios typical in video coding applications: conversational services and internet protocol television (IPTV). In the conversational scenario two QCIF video sequences, namely *Foreman* and *Coastguard*, have been coded at 64 kbps and 15 fps temporal resolution. Conversely, in the IPTV scenario two CIF video sequences, namely *Soccer* and *Hall Monitor*, and two 625 standard definition (SD) video sequences, namely *Rugby* and *Mobile & Calendar*,[2] have been coded at, respectively, 256 kbps–30 fps and 4M bps–25 fps. For all the tested video sequences the H.264/AVC reference software model (version JM12.3 [25]) has been used with baseline profile for the QCIF and CIF video sequences and main profile for the SD sequences. The remaining source coding parameters for the H.264/AVC encoder are listed in Table V.

Each coded frame is partitioned into slices, where each slice contains a horizontal row of macroblocks. Each coded slice is then packetized according to the real-time transfer protocol (RTP) specifications [37]. The simulated error prone channel drops coded packets according to a packet loss rate in the range [0.1, 20]. The error patterns have been generated using a two states Gilbert's model [27] with average burst length of three packets.

The reduced-reference quantities (i.e., $\hat{\mu}_{\hat{\mathbf{M}}_n^i}$ and $\hat{\sigma}_{\hat{\mathbf{M}}_n^i}$) are gathered into two sequences each composed by frames of dimensions $R \times C$, where R and C are, respectively, the number of rows and columns of a frame measured in terms of macroblock units. The two sequences relative to $\hat{\mu}_{\hat{\mathbf{M}}_n^i}$ and $\hat{\sigma}_{\hat{\mathbf{M}}_n^i}$ are then lossy encoded by a zero-motion, transformed domain, differential pulse code modulation (DPCM) scheme which adopts the context adaptive binary arithmetic coding (CABAC) as entropy encoder and a deadzone uniform quantizer equal to the one adopted for the H.264/AVC video coding standard [9]. The bit-rate required for encoding the RR information for each tested sequence is listed in Table VI together with the used quantization parameter (QP) values for both $\hat{\mu}_{\hat{\mathbf{M}}_n^i}$ and $\hat{\sigma}_{\hat{\mathbf{M}}_n^i}$. Distributed source coding tools might be employed in order to take advantage of the correlation between the quantities to be encoded (i.e., $\hat{\mu}_{\hat{\mathbf{M}}_n^i}$ and $\hat{\sigma}_{\hat{\mathbf{M}}_n^i}$) and the side information available only at the decoder ($\tilde{\mu}_{\hat{\mathbf{M}}_n^i}$ and $\tilde{\sigma}_{\hat{\mathbf{M}}_n^i}$). This is left to future investigations.

[2]The original file names for these video sequences are, respectively, *src9_ref_625* and *src10_ref_625* and have been downloaded from [36].

TABLE VI
BIT-RATES REQUIRED FOR RR INFORMATION ENCODING

Sequence name	Bitrate [kbps]	$QP(\hat{\mu}_{\hat{\mathbf{M}}_n^i})$	$QP(\hat{\sigma}_{\hat{\mathbf{M}}_n^i})$
Foreman	3.89	25	40
Coastguard	2.72	25	40
Soccer	21.37	25	40
Hall Monitor	9.13	25	40
Rugby	22.74	25	40
Mobile & Calendar	10.70	25	40

B. Comparison Setup

We compare the proposed no-reference channel induced distortion estimation algorithm with other three methods described in the literature: the FP and QP methods as proposed by Reibman *et al.* in [10] and the method by Yamada *et al.* described in [16]. The algorithms in [10] have been originally proposed to be used with MPEG-2 encoded video material, and they are adapted here to work on H.264/AVC video bitstreams. In the FP method, the authors state that for missing macroblocks, the seven parameters needed for computing the MSE distortion (corresponding to macroblock variances, means, horizontal and vertical correlations) are estimated from their neighbors. In our implementation we consider as neighbors the four macroblocks surrounding the missing one. If all these macroblocks are also missing, we estimate the parameters from the co-located macroblock in the previous frame. The QP method needs an estimate of a parameter that corresponds to the innovation in the MSE due to loosing a slice in a frame at time n and concealing it with the same slice in a frame at time $n-t$. In [10], the value of this parameter is obtained with training over a set of eight video sequences. Here we use one set of eight video sequences for each considered spatial resolution. For the QCIF and CIF spatial resolutions we trained the QP system on the following sequences: *Akiyo, Carphone, Claire, Mobile & Calendar, Mother & Daughter, News, Salesman,* and *Table Tennis*. For the SD scenario we use the sequences "*src_x_ref_625*", where x spans from 1 to 8, which are available for download from the VQEG web site [36].

In the method by Yamada *et al.* we adjusted the two thresholds: $\mathrm{Th}_{\mathrm{mv}}$ and Th_L over the same sets of training video sequences used for the QP method. Following the guidelines in [16] and under the aforementioned experimental conditions, the values for $\mathrm{Th}_{\mathrm{mv}}$ and Th_L have been set to, respectively, 3 and

$$\mathrm{SSIM}_{\mathrm{RR}}(\hat{\mathbf{M}}_n^i, \tilde{\mathbf{M}}_n^i) = \frac{\left(2\hat{\mu}_{\hat{\mathbf{M}}_n^i}\mu_{\tilde{\mathbf{M}}_n^i} + C_1\right)\cdot\left(2\hat{\sigma}_{\hat{\mathbf{M}}_n^i,\tilde{\mathbf{M}}_n^i} + C_2\right)}{\left(\hat{\mu}^2_{\hat{\mathbf{M}}_n^i} + \mu^2_{\tilde{\mathbf{M}}_n^i} + C_1\right)\cdot\left(\hat{\sigma}^2_{\hat{\mathbf{M}}_n^i} + \sigma^2_{\tilde{\mathbf{M}}_n^i} + C_2\right)} \tag{30}$$

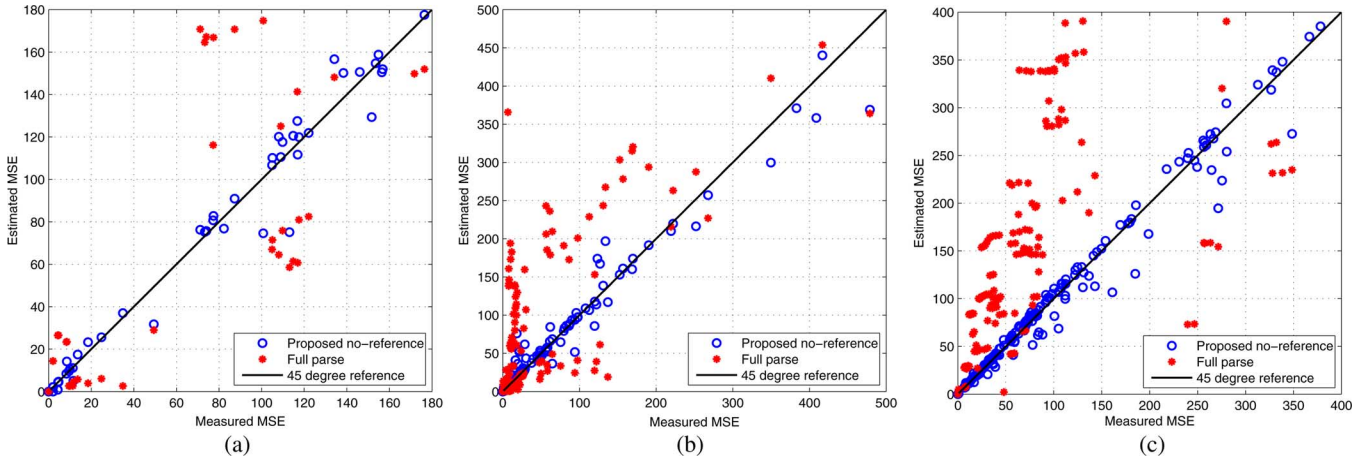

Fig. 5. Examples of frame level scatter plots for some of the tested sequences when the PLR is equal to 2.2%. (a) *Foreman* QCIF. (b) *Soccer* CIF. (c) *Mobile & Calendar* SD.

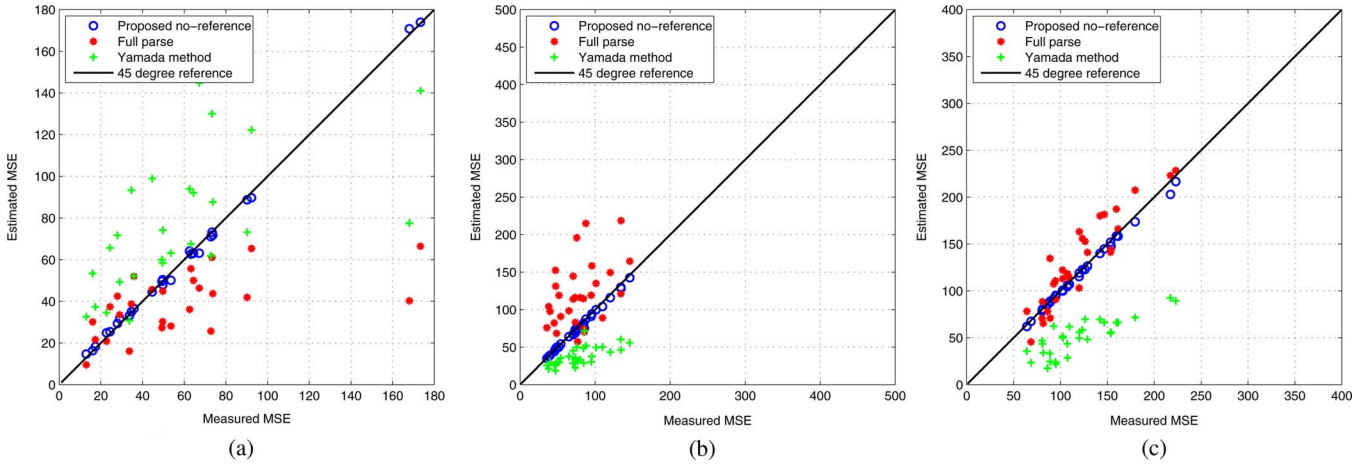

Fig. 6. Examples of sequence level scatter plots for some of the tested sequences when the PLR is equal to 2.2%. (a) *Foreman* QCIF. (b) *Soccer* CIF. (c) *Mobile & Calendar* SD.

5. With these values the polynomial fitting gives the following expression:

$$\hat{D}_{\text{seq}} = 0.0027 \cdot x^2 - 0.9057 \cdot x + 65.53 \qquad (31)$$

where D_{seq} is the average distortion at the sequence level and x represents the number of macroblocks over the entire sequence judged as ineffectively concealed by the metrics proposed in [16].

As for the proposed reduced-reference quality assessment method, we compare our estimated SSIM values with the ones calculated in full-reference mode as in [33].

C. Experimental Results and Discussion

We simulated the transmission of the test sequences over 30 channel realizations for each considered PLR value. In order to quantify the accuracy of both the proposed no-reference and reduced-reference channel induced distortion estimation algorithms we measured the Pearson's correlation coefficient between, respectively, the estimated and actual MSE distortion, i.e., $\hat{D}$ and D and the SSIM_{RR} and SSIM_{FR}. In the following we present our results distinguishing between the no-reference and reduced-reference cases.

1) No-Reference Quality Monitoring: We measure the accuracy of the estimated distortion at macroblock (MB), frame and sequence (Seq) level. As a first example, Table VII shows the values of the correlation coefficient computed for all the tested algorithms for the *Foreman* sequence. We notice that NORM performs better than the other tested techniques. In fact, at the macroblock level, the correlation coefficient is greater than 0.81, and compares favorably with the the FP method in [10], which attains a correlation greater than 0.61. The same observation holds at the frame and sequence level, where the proposed algorithm achieves a correlation greater than 0.90. The performance achieved for the *Foreman* video sequence is confirmed on other tested video sequences (Tables VIII–X). We observe that the complexity of motion affects the accuracy of the estimate provided by the NORM algorithm. In fact, for sequences characterized by complex motion, like *Rugby*, the correlation between the true and estimated MSE at the macroblock level is between 0.76–0.83. A higher correlation is achieved for sequences characterized by simpler motion, e.g., *Foreman* (0.81–0.93) and *Hall*

TABLE VII
CORRELATION ρ COEFFICIENT BETWEEN $\hat{D}$ AND D AT DIFFERENT LEVELS OF GRANULARITY FOR THE *Foreman* VIDEO SEQUENCE

	NORM			Full parse			Quick parse		Yamada et al.
PLR [%]	MB	Frame	Seq	MB	Frame	Seq	Frame	Seq	Seq
0.1	0.94	0.96	0.99	0.70	0.83	0.85	0.72	0.84	0.94
0.4	0.93	0.97	0.98	0.69	0.83	0.85	0.70	0.85	0.93
0.7	0.94	0.95	0.99	0.68	0.83	0.85	0.71	0.86	0.89
1.0	0.93	0.97	0.99	0.68	0.83	0.85	0.70	0.88	0.87
1.3	0.93	0.97	0.99	0.67	0.82	0.85	0.71	0.85	0.87
1.6	0.93	0.97	0.99	0.67	0.82	0.84	0.71	0.81	0.86
1.9	0.93	0.97	0.98	0.67	0.81	0.85	0.69	0.82	0.83
2.2	0.93	0.96	0.98	0.67	0.80	0.84	0.69	0.82	0.82
2.5	0.92	0.96	0.98	0.66	0.79	0.84	0.69	0.82	0.81
3	0.87	0.92	0.94	0.64	0.75	0.85	0.66	0.79	0.78
5	0.85	0.88	0.91	0.63	0.76	0.85	0.64	0.75	0.76
10	0.83	0.89	0.90	0.63	0.75	0.84	0.65	0.72	0.74
20	0.81	0.87	0.90	0.61	0.75	0.83	0.65	0.68	0.73

TABLE VIII
CORRELATION ρ COEFFICIENT BETWEEN $\hat{D}$ AND D AT DIFFERENT LEVELS OF GRANULARITY FOR THE *Soccer* VIDEO SEQUENCE

	NORM			Full parse			Quick parse		Yamada et al.
PLR [%]	MB	Frame	Seq	MB	Frame	Seq	Frame	Seq	Seq
0.1	0.89	0.96	0.98	0.65	0.81	0.88	0.64	0.79	0.88
0.4	0.90	0.96	0.98	0.64	0.79	0.87	0.63	0.78	0.86
0.7	0.90	0.96	0.97	0.64	0.80	0.86	0.62	0.79	0.86
1.0	0.89	0.96	0.98	0.62	0.79	0.86	0.62	0.78	0.85
1.3	0.91	0.96	0.97	0.62	0.79	0.85	0.61	0.78	0.84
1.6	0.89	0.95	0.98	0.60	0.79	0.85	0.60	0.77	0.84
1.9	0.89	0.95	0.97	0.59	0.77	0.83	0.58	0.78	0.83
2.2	0.90	0.95	0.97	0.58	0.77	0.84	0.57	0.78	0.83
2.5	0.90	0.95	0.97	0.59	0.75	0.83	0.56	0.79	0.81
3	0.88	0.94	0.96	0.56	0.73	0.81	0.52	0.79	0.82
5	0.86	0.93	0.97	0.55	0.70	0.76	0.52	0.78	0.76
10	0.83	0.94	0.96	0.53	0.71	0.73	0.51	0.74	0.77
20	0.79	0.91	0.95	0.54	0.68	0.71	0.50	0.74	0.74

Monitor (0.84–0.94). The accuracy of the estimated MSE depends also on the PLR. In fact, the correlation tends to decrease at PLR higher than 3%, especially when the motion is complex. As a visual comparison, we report also the scatter plots at the frame and sequence level showing the estimated vs. actual distortion. Fig. 5 illustrates the scatter plot both for the proposed NORM algorithm and FP over the sequences *Foreman*, *Soccer*, and *Mobile & Calendar* when the PLR is equal to 2.2%. We notice that the estimate provided by the proposed NORM algorithm accurately fits the ground truth data with only a small fraction of outliers. Conversely, the estimate provided by the FP method deviates from the ground truth especially for MSE values higher than 50 in the *Soccer* sequence. The observed results at the frame level hold also at the sequence level as shown in Fig. 6. We also depict the results obtained with the method by Yamada *et al.* described in [16]. Both methods show a significant deviation with respect to the ground truth in terms of variance and bias.

We now discuss the reasons underlying the better accuracy obtained by NORM with respect to the other tested methods. First, NORM explicitly takes advantage of knowledge of the predictor $\tilde{\mathbf{P}}_n^i$ computed by motion-compensated temporal concealment. As a matter of fact, $\tilde{\mathbf{P}}_n^i$ represents a good estimate of the missing macroblock $\hat{\mathbf{M}}_n^i$ and it can be reliably used to estimate the frequency domain properties of the missing macroblock, as needed by the model in (21). Conversely, the FP method implicitly assumes a simple zero-motion temporal concealment, neglecting the actual concealment strategy carried out at the decoder. This is due to the fact that FP is designed to work in the DCT domain avoiding motion-compensation. The computational complexity of the FP algorithm is similar to that of NORM (see Section III-H): the cost of computing the temporal error propagation is essentially equivalent, and grows linearly with the frame size. In terms of the distortion introduced by the loss of a packet, FP does not require to compute the FFT, hence the term $cB^2 \log_2 B$ can be dropped. The most significant difference in terms of complexity is related to the computation of the data fed in input to the two algorithms. In fact NORM requires a full decoding of the bitstream up to the pixel domain, whereas FP requires only entropy decoding. Nevertheless, if quality monitoring has to be performed at the receiver node, the sequence needs to be fully decoded anyway.

As far as the method in [16] is concerned, the quadratic fitting in (31) is a sort of blind-box approach that ignores the specific processing carried out at the decoder. Moreover when the characteristics of the decoded sequence do not match well those of

TABLE IX
CORRELATION ρ COEFFICIENT BETWEEN $\hat{D}$ AND D AT DIFFERENT LEVELS OF GRANULARITY FOR THE *Rugby* VIDEO SEQUENCE

	NORM			Full parse			Quick parse		Yamada et al.
PLR [%]	MB	Frame	Seq	MB	Frame	Seq	Frame	Seq	Seq
0.1	0.83	0.97	0.98	0.63	0.83	0.90	0.66	0.81	0.85
0.4	0.82	0.96	0.98	0.62	0.83	0.92	0.65	0.78	0.83
0.7	0.82	0.96	0.99	0.61	0.82	0.92	0.67	0.79	0.82
1.0	0.81	0.96	0.99	0.60	0.83	0.90	0.66	0.77	0.82
1.3	0.81	0.96	0.99	0.61	0.81	0.90	0.66	0.74	0.79
1.6	0.81	0.96	0.98	0.60	0.82	0.88	0.69	0.76	0.78
1.9	0.80	0.96	0.99	0.56	0.80	0.88	0.65	0.76	0.78
2.2	0.80	0.96	0.98	0.57	0.79	0.87	0.64	0.74	0.78
2.5	0.80	0.96	0.97	0.56	0.77	0.88	0.64	0.74	0.76
3	0.78	0.94	0.95	0.55	0.77	0.88	0.61	0.72	0.81
5	0.78	0.91	0.94	0.55	0.76	0.87	0.60	0.72	0.77
10	0.77	0.90	0.95	0.55	0.75	0.88	0.59	0.67	0.75
20	0.76	0.89	0.93	0.54	0.73	0.87	0.56	0.67	0.69

TABLE X
CORRELATION ρ COEFFICIENT BETWEEN $\hat{D}$ AND D AT DIFFERENT LEVELS OF GRANULARITY FOR ALL THE TESTED VIDEOS AND PLRS

		NORM			Full parse			Quick parse		Yamada et al.
Sequence name		MB	Frame	Seq	MB	Frame	Seq	Frame	Seq	Seq
Foreman	min	0.81	0.87	0.90	0.61	0.75	0.83	0.64	0.68	0.73
	max	0.93	0.96	0.99	0.70	0.83	0.85	0.72	0.88	0.94
Coastguard	min	0.81	0.87	0.91	0.61	0.72	0.86	0.57	0.77	0.85
	max	0.92	0.97	0.99	0.68	0.85	0.95	0.70	0.87	0.96
Soccer	min	0.79	0.91	0.95	0.53	0.68	0.71	0.50	0.74	0.74
	max	0.91	0.96	0.98	0.65	0.81	0.88	0.64	0.79	0.88
Hall Monitor	min	0.84	0.90	0.94	0.59	0.72	0.80	0.60	0.77	0.81
	max	0.94	0.96	0.99	0.69	0.86	0.94	0.68	0.85	0.94
Rugby	min	0.76	0.89	0.93	0.54	0.73	0.87	0.56	0.67	0.69
	max	0.83	0.97	0.99	0.63	0.83	0.92	0.69	0.81	0.85
Mobile & Calendar	min	0.82	0.89	0.91	0.57	0.74	0.89	0.57	0.83	0.73
	max	0.91	0.95	0.99	0.69	0.85	0.94	0.72	0.88	0.81

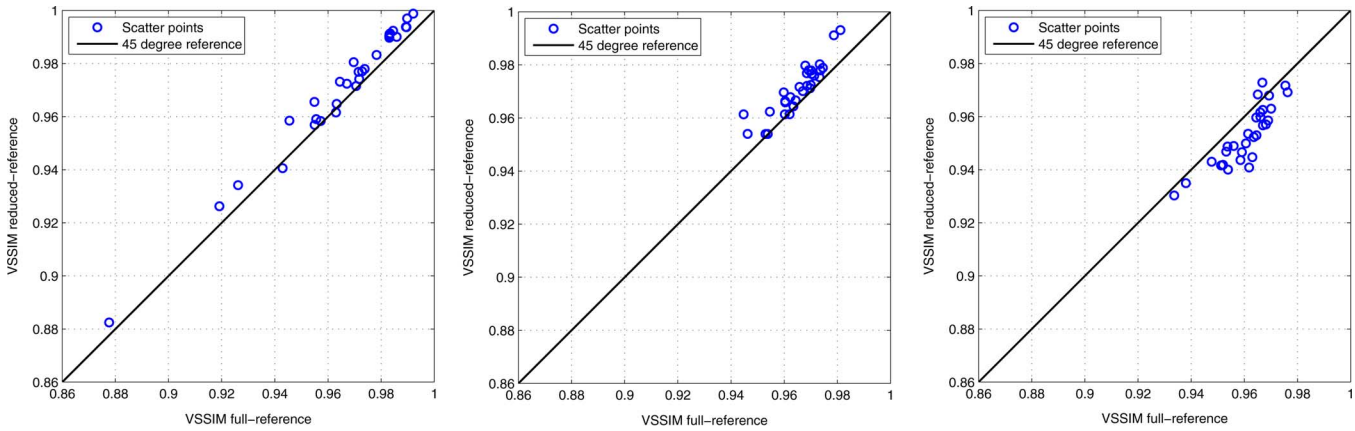

Fig. 7. Sequence level SSIM scatter plots between $\mathrm{SSIM}_{\mathrm{FR}}(\hat{X}, \bar{X})$ and $\mathrm{SSIM}_{\mathrm{RR}}(\hat{X}, \bar{X})$ when the PLR is equal to 2.2%. (a) *Foreman* QCIF. (b) *Soccer* CIF. (c) *Mobile & Calendar* SD.

the sequences used for training, the predicted distortion can be significantly different from the actual one. Yet, this method has an overall complexity comparable with our approach since it requires full decoding of the bitstream in order to determine the number of inefficiently concealed macroblocks.

2) Reduced-Reference Quality Monitoring: The proposed NORM algorithm attains high correlation values also at the macroblock level. Therefore we decided to leverage the output of NORM in the context of a reduced-reference objective quality assessment algorithm. We have measured the Pearson's correlation values for the video structural similarity metric [33] between $\mathrm{SSIM}_{\mathrm{RR}}$ and $\mathrm{SSIM}_{\mathrm{FR}}$. The correlation values are shown in Table XI. We notice that a good correlation exists (higher than 0.81) between $\mathrm{SSIM}_{\mathrm{RR}}$ and $\mathrm{SSIM}_{\mathrm{FR}}$. As a graphical example, we also illustrate the scatter plots for the *Foreman*, *Soccer*, and *Mobile & Calendar* video sequences [Fig. 7(a)–(c)].

TABLE XI
SEQUENCE LEVEL CORRELATION VALUES BETWEEN $\mathrm{SSIM}_{\mathrm{FR}}$ AND $\mathrm{SSIM}_{\mathrm{RR}}$

PLR [%]	Sequence name					
	Foreman	*Coastguard*	*Soccer*	*Hall Monitor*	*Rugby*	*Mobile & Calendar*
0.1	0.99	0.95	0.94	0.99	0.91	0.90
0.4	0.98	0.94	0.93	0.98	0.90	0.88
0.7	0.94	0.96	0.93	0.96	0.88	0.88
1.0	0.94	0.94	0.92	0.97	0.88	0.88
1.3	0.95	0.96	0.91	0.95	0.87	0.90
1.6	0.98	0.93	0.92	0.93	0.86	0.88
1.9	0.98	0.92	0.91	0.97	0.85	0.87
2.2	0.97	0.91	0.90	0.86	0.83	0.86
2.5	0.94	0.91	0.87	0.97	0.81	0.87

VI. CONCLUSION

In this paper we have presented NORM, a NO-Reference video quality Monitoring algorithm that estimates the distortion due to channel errors in video sequences coded with H.264/AVC. The proposed method explicitly models the distortion due to the lack of motion vectors, prediction residuals and temporal propagation. The provided estimate shows a good accuracy also at the macroblock level and compare favorably with other methods proposed in the literature. NORM can be easily embedded in any H.264/AVC complaint decoder. Furthermore, in order to overcome the poor correlation that sometimes exists between MSE distortion and MOS collected in subjective tests, we have exploited the output of NORM to feed an algorithm that computes a reduced-reference approximation of the SSIM metrics. In this case we have shown that the SSIM metrics computed in reduced-reference mode is well correlated with the SSIM computed in full-reference mode. As a future investigation, we will consider to exploit NORM to compute no-reference objective quality metrics based either on regional attention [11] or perceptual foveation [32].

REFERENCES

[1] M. Naccari, M. Tagliasacchi, F. Pereira, and S. Tubaro, "No-reference modeling of the channel induced distortion at the decoder for H.264/AVC video coding," in *Proc. Int. Conf. Image Processing*, San Diego, CA, Oct. 2008.

[2] K. Stuhlmüller, N. Färber, M. Link, and B. Girod, "Analysis of video transmission over lossy channels," *IEEE J. Select. Areas Commun.*, vol. 18, no. 6, pp. 1012–1032, Jun. 2000.

[3] 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, Oct. 1999.

[4] I. E. G. Richardson, *Video Codec Design*. New York: Wiley, 2002.

[5] M. Claypool and J. Tanner, "The effects of jitter on the perceptual quality of video," in *Proc. ACM Multimedia*, Orlando, FL, Nov. 1999.

[6] Y.-C. Chang, T. Carney, S. A. Klein, D. G. Messerschmitt, and A. Zakhor, B. E. Rogowitz and T. N. Pappas, Eds., "Effects of temporal jitter on video quality: Assessment using psychophysical and computational modeling methods," in *Proc. Society of Photo-Optical Instrumentation Engineers (SPIE) Conf. Series*, Jul. 1998, vol. 3299, pp. 173–179.

[7] S. Winkler, "Video quality and beyond," in *Proc. Eur. Signal Processing Conf.*, Poznań, Poland, Sep. 2007.

[8] S. Winkler and P. Mohandas, "The evolution of video quality measurement: From PSNR to hybrid metrics," *IEEE Trans. Broadcast*, to be published.

[9] 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, Jul. 2003.

[10] 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, Apr. 2004.

[11] U. Engelke, V. X. Nguyen, and H.-J. Zepernick, "Regional attention to structual degradation for perceptual image quality metric design," in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing*, Las Vegas, NV, Apr. 2008.

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

[13] R. Zhang, S. L. Regunathan, and K. Rose, "Video coding with optimal inter/intra-mode switching for packet loss resilience," *IEEE J. Select. Areas Commun.*, vol. 18, no. 6, pp. 966–976, Jun. 2000.

[14] H. Yang and K. Rose, "Advances in recursive per-pixel end-to-end distortion estimation for robust video coding in H.264/AVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 7, pp. 845–856, Jul. 2007.

[15] T. Shu, J. Apostolopoulos, and R. Guérin, "Real-time monitoring of video quality in IP networks," in *Proc. Int. Workshop Network and Operating System Support for Digital Audio and Video*, Stevenson, WA, Jun. 2005.

[16] T. Yamada, Y. Miyamoto, and M. Serizawa, "No-reference video quality estimation based on error-concealment effectiveness," in *IEEE Packet Video*, Lausanne, Switzerland, Nov. 2007.

[17] 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. 341–355, Apr. 2006.

[18] 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. Int. Conf. Image Processing*, Atlanta, GA, Oct. 2006.

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

[20] A. Ichigaya, M. Kurozumi, N. Hara, Y. Nishida, and E. Nakasu, "A method of estimating coding PSNR using quantized DCT coefficients," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 2, pp. 251–259, Feb. 2006.

[21] Z. He, J. Cai, and C. W. Chen, "Joint source channel rate-distortion analysis for adaptive mode selection and rate control in wireless video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 6, pp. 511–523, Jun. 2002.

[22] P. List, A. Joch, J. Lainema, G. Bjøntegaard, and M. Karczewiczt, "Adaptive deblocking filter," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 614–619, Jul. 2003.

[23] G. J. Sullivan, T. Wiegand, and K.-P. Lim, Joint Model Reference Encoding Methods and Decoding Concealment Methods, Joint Video Team (JVT), Tech. Rep. JVT-I049, Sep. 2003.

[24] D. Agrafiotis, D. R. Bull, and C. N. Canagarajah, "Enhanced error concealment with mode selection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 8, pp. 960–973, Aug. 2006.

[25] Joint Video Team (JVT), *H.264/AVC Reference Software Version JM12.3*. [Online]. Available: http://iphome.hhi.de/suehring/tml/download/.

[26] Y. J. Liang, J. G. Apostolopoulos, and B. Girod, "Analysis of packet loss for compressed video: Does burst-length matter?," in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing*, Hong Kong, Apr. 2003.

[27] E. N. Gilbert, "Capacity of a burst-noise channel," *Bell Syst. Tech. J.*, vol. 39, pp. 1253–1266, Sep. 1960.

[28] 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, Dec. 2006.

[29] Y. Wang, Z. Wu, and J. M. Boyce, "Modeling of transmission-loss-induced distortion in decoded video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 6, pp. 716–732, Jun. 2006.

[30] *Final Draft International Standard*, ISO-IEC FDIS 14 496-10, Mar. 2003, Information Technology—Coding of audio-visual objects—Part 10: advanced video coding, ITU-T.
[31] A. Secker and D. Taubman, "Highly scalable video compression with scalable motion coding," *IEEE Trans. Image Process.*, vol. 13, no. 8, pp. 1029–1041, Aug. 2004.
[32] Z. Wang and A. C. Bovik, "Embedded foveation image coding," *IEEE Trans. Image Process.*, vol. 10, no. 10, pp. 1397–1410, Oct. 2001.
[33] Z. Wang, L. Lu, and A. C. Bovik, "Video quality assessment based on structural distortion measure," *Signal Process.: Image Commun.*, vol. 19, no. 2, pp. 121–132, Feb. 2004.
[34] K. Seshadrinathan and A. C. Bovik, "A structural similarity metric for video based on motion models," in *Proc. Int. Conf. Acoustics, Speech, and Signal Processing*, Honolulu, HI, Apr. 2007.
[35] Z. Wang, H. R. Sheikh, and A. C. Bovik, *The Handbook of Video Databases: Design and Applications*. Boca Raton, FL: CRC, 2003, ch. 41: Objective video quality assessment, pp. 1041–1078.
[36] The Video Quality Expert Group Web Site. [Online]. Available: http://www.its.bldrdoc.gov/vqeg.
[37] S. Wenger, "H.264/AVC over IP," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 645–656, Jul. 2003.

Matteo Naccari (S'08) was born in Como, Italy, in 1979. He is pursing the Ph.D. degree at the "Dipartimento di Elettronica e Informazione", Politecnico di Milano, Milano, Italy, working in the Image and Sound Processing Group (ISPG), with a scholarship granted by ST Microelectronics.

His research interests are mainly concerned in the video coding area where he works on video transcoding architectures, error resilience video coding, and automatic quality monitoring in video content delivery. He actively collaborates with the Image Group at the Instituto Superior Técnico Lisbon, Portugal, with the Telecommunications and Signal Processing Group at the Università degli Studi di Udine, and with the Advanced System Technology Group of ST Microelectronics.

Marco Tagliasacchi (M'06) was born in 1978. He received the "Laurea" degree (*Cum Laude*) in computer engineering and the Ph.D. degree in electrical engineering and computer science from the "Politecnico di Milano", Milano, Italy, in 2002 and 2006, respectively.

Since February 2007, he has been an Assistant Professor at the "Dipartimento di Elettronica e Informazione"—"Politecnico di Milano". He authored more than 50 papers in international journals and conferences. His research interests include video coding (scalable video coding, distributed video coding, error resilient coding, non-normative tools in video coding standards), applications of compressive sensing (audio and image tempering localization), robust classification of acoustic events, and localization of acoustic sources through microphone arrays. He has been actively involved in several publicly and privately funded projects.

Dr. Tagliasacchi is a member of the IEEE Multimedia Signal Processing Technical Committee for the term 2009–2011.

Stefano Tubaro (M'01) was born in Novara, Italy, in 1957. He received the electronic engineering degree at the "Politecnico di Milano", Milano, Italy, in 1982.

He then joined the "Dipartimento di Elettronica e Informazione" of the "Politecnico di Milano", first as a researcher of the National Research Council, then (in November 1991) as an Associate Professor, and from December 2004 as a Full Professor. In the first years of activities, he worked on problems related to speech analysis; motion estimation/compensation for video analysis/coding; and vector quantization applied to hybrid video coding. In the past few years, his research interests have focused on image and video analysis for the geometric and radiometric modeling of 3-D scenes as well as advanced algorithms for video coding and sound processing. He authored more than 150 scientific publications on international journals and congresses. He co-authored two books on digital processing of video sequences. He is also a co-author of several patents relative to image processing techniques. He coordinates the research activities of the Image and Sound Processing Group (ISPG) at the "Dipartimento di Elettronica e Informazione" of the "Politecnico di Milano" that is involved in several research programs funded by industrial partners, the Italian Government, and by the European Commission.