Iterative Side-Information Generation in a Mixed Resolution Wyner–Ziv Framework

Bruno Macchiavello, Debargha Mukherjee, *Senior Member, IEEE,* and Ricardo L. de Queiroz, *Senior Member, IEEE*

***Abstract*—We propose a mixed resolution framework based on full resolution key frames and spatial-reduction-based Wyner–Ziv coding of intermediate nonreference frames. Improved rate-distortion performance is achieved by enabling better side-information generation at the decoder side and better rate-allocation at the encoder side. The framework enables reduced encoding complexity by low resolution encoding of the nonreference frames, followed by Wyner–Ziv coding of the Laplacian residue. The quantized transform coefficients of the residual frame are mapped to cosets without the use of a feedback channel. A study to select optimal coding parameters in the creation of the memoryless cosets is made. Furthermore, a correlation estimation mechanism that guides the parameter choice process is proposed. The decoder first decodes the low resolution base layer and then generates a super-resolved side-information frame at full resolution using past and future key frames. Coset decoding is carried using side-information to obtain a higher quality version of the decoded frame. Implementation results are presented for the H.264/AVC codec.**

***Index Terms*—Distributed video coding, reversed-complexity, scalable video coding, side-information generation.**

I. Introduction

DISTRIBUTED SOURCE CODING (DSC) has its roots in the works of Slepian and Wolf [1] for the lossless case and Wyner and Ziv [2] for the lossy case. It is the focus of different kinds of video coding schemes [3]–[12]. Currently, digital video standards have highly complex encoders [13], mainly due to motion estimation (ME). On the other hand, the decoder complexity is low, following a broadcast-oriented model. However, for real-time encoding with limited power the distributed video coding (DVC) paradigm is preferred. DVC operates with reverse complexity, which means that the encoder complexity is shifted to the decoder.

In realistic mobile video communications, it may not be necessary for the video encoder to always operate in a reversed-complexity mode. This mode may be turned on only when available battery power drops. Besides, while reducing complexity is important, it should not be achieved at a substantial cost in bandwidth, i.e., the amount of complexity reduction should be adaptive in the interest of a better rate-distortion (RD) trade-off. Furthermore, video communicated from one mobile device may be played back in real-time at another mobile device. Hence, the decoder must support a mode where at least a low quality version of the bit-stream can be decoded with low complexity. Offline processing may be conducted for retrieving the higher quality version. If not the case, a transcoder inserted into the network may enable low complexity at both ends [14]. While many works in DVC [4], [11], [15] require a feedback channel, we consider the more practical blind case, where the decoder is not necessarily required to attempt distributed decoding immediately after reception. In most real-time communication situations, the latency constraints may be too stringent, anyway, to make application-level feedback effective. Therefore, our framework does not employ a feedback channel. This scenario allows DVC coding across all power-constrained devices. However, it increases the difficulty in choosing the correct WZ coding parameters in order to approach the Slepian–Wolf frontier. With a feedback channel, the decoder may ask for more information depending on the quality of the side-information (SI) generated. In the blind case, this is not possible. Therefore, estimation of the correlation noise and the signal statistics is necessary in order to send the exact amount of information. We propose a method for correlation statistics estimation and coding parameter selection for the memoryless cosets [16].

Typically, reversed-complexity schemes use periodic INTRA-coded key frames with multiple motion-free Wyner–Ziv (WZ) frames in between [4]. Then, without ME, the encoder is substantially less complex than the decoder. However, this framework limits the RD efficiency, due to occlusions and complex motion. In order to generate an accurate enough SI at the decoder, frequent key frames are required. One way to improve efficiency is to replace key frames (I) with conventional P-frame or B-frame. In that case, the nonkey frames should be nonreference frames to avoid drifting errors. However, the quality of SI would still limit the performance. In order to improve the quality of the SI, a variation of this class of methods [5], [17] transmits an auxiliary information for each coded block to help the block matching process. Nevertheless, the rate needed for the hash is often prohibitive. Another approach [18] uses a highly compressed version of each WZ frame, with zero-motion vectors, as a frame-hash to improve the SI. However, the performance of

Manuscript received January 9, 2008; revised June 18, 2008. First version published July 7, 2009; current version published September 30, 2009. This work was supported by Hewlett-Packard. This paper was recommended by Associate Editor J. Boyce.

B. Macchiavello and R. L. de Queiroz are with the Departamento de Engenharia Eletrica, Universidade de Brasilia, Brasilia, Brazil (e-mail: bruno@image.unb.br; queiroz@ieee.org).

D. Mukherjee is with Hewlett-Packard Laboratories, Palo Alto, CA 94304-1126 USA (e-mail: debargha.mukherjee@hp.com).

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/TCSVT.2009.2026820

all these methods, in the no-feedback-channel case, is heavily dependent on the accuracy of the rate-allocation mechanism. Typically, there is no information shared between the hash layer and the WZ bit-stream. Some studies have addressed the rate-allocation problem for the motion-free WZ architecture when a feedback channel is not employed [19]–[21]. A rate-allocation mechanism have been proposed for a pixel-domain [19] and transform domain [20] distributed video codec. In both cases, a difference between pixel values of the original frame and a reference frame is used to predict the bit error probabilities. Also, an adaptive rate-allocation technique for a multiuser video coding system has been presented [21].

An alternative is to use spatial-reduction-based DVC [22]–[24]. We call this system mixed resolution DC or MR-DVC. In the particular case [22] that we explore in this paper, the key frames are conventional I, P, or reference-B frames coded at full resolution. The intermediate nonreference WZ frames are coded in two layers: a base layer coded at reduced resolution, followed by an enhancement layer that uses WZ coding of the Laplacian residue. The low resolution (LR) base layer of the WZ frames can be regarded as an efficiently compressed hash for the frame [18]. MR-DVC [22] can be implemented as an optional coding mode to any existing video codec standard [13], [25]. At the decoder, if full resolution key frames along with only the low resolution base layers of the WZ frames are decoded, we would have achieved low complexity but low quality decoding. For decoding the enhancement layer, we propose a motion-based multihypothesis SI generation method that uses information from full resolution key frames along with information from the LR base layer. Other works have used iterative SI generation techniques [26]–[28]. All of them assume key frames are intra coded and the intermediate frames are entirely WZ coded. In [27], previously decoded bitplanes are used to improve SI. In [26] the SI is generated by aggressively replacing LR blocks by blocks from the key frames.

In this paper, a method for estimating the correlation statistics is proposed. An optimal parameter choice mechanism in the creation of memoryless cosets [23] is presented. The decoded WZ frame is obtained iteratively by alternating multiframe super-resolution reconstruction and memoryless coset decoding processes. More complex channel codes may improve the performance. However, it has been recently shown [29] that without a feedback channel, when the likelihood of channel mismatch is high, memoryless coset codes can be more robust than other codes. In an independently developed work [30], spatial resolution reduction has also been used as a means for complexity reduction. There, the LR base layer uses zero-motion vector. Our approach has a moderate complexity reduction target and allows for regular motion search at reduced resolution, thereby substantially improving RD performance. Note that the correlation estimation and parameter choice can be applied with minor modifications to other frameworks [18], [30]. DVC with spatio-temporal scalability has been previously addressed [9]. The use of spatial scalability for coding the key frames may degrade the RD performance at full spatial resolution. Also, layered Wyner–Ziv codecs with similar results to the MPEG-4/H.264 FGS have been proposed [7], [8] based on recent theoretical results on successive refinement of Gaussian sources for WZ coding [31].

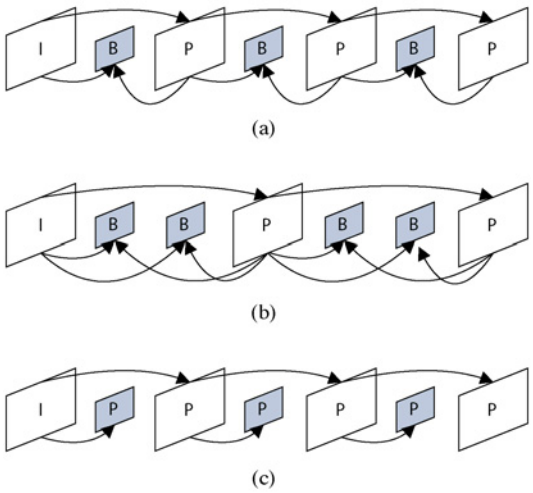

Fig. 1. Different uses of NRWZ frames. (a) One B-like NRWZ frame between reference frames. (b) Two B-like NRWZ frames. (c) One P-like NRWZ frames.

For the SI generation problem, an RD analysis of motion-based SI estimators has been addressed [32], [33]. Another work [34] uses mode-aided motion compensation to generate the SI. Such studies are relevant to our spatial-reduction framework as well.

The paper is organized as follows. The framework is described in Section II. In Section III, the generation of the SI is introduced and in Section IV we show how the parameters for the coset generation are chosen based on the estimated variances. We describe how the variances are estimated in Section V. Finally, the implementation results of the proposed method using the H.264/AVC codec are presented in Section VI, followed by the conclusions in Section VII.

II. MR-DVC Framework

Our scalable framework is based on nonreference WZ frames (NRWZ) [22], [23]. The complexity reduction is applied only to nonreference frames. Since the reference frames are coded exactly as in a conventional codec as I-frame, P-frame, or reference B-frame, there are no drifting errors. This feature also enables the receiver to immediately playback a lower quality version of the video with a conventional decoder, leaving full decoding of the NRWZ frames to offline processing. The framework does not limit the number of nonreference frames between the reference frames. Ideally, the number of NRWZ frames can be varied dynamically based on the complexity reduction required and on the target quality. Fig. 1 depicts typical frame configurations. In Fig. 1(a), the B frames of a conventionally coded sequence have been converted into NRWZ B-like frames, while Fig. 1(b) shows a similar sequence with lower complexity. Fig. 1(c) shows a low delay case where NRWZ frames are used like P-frames.

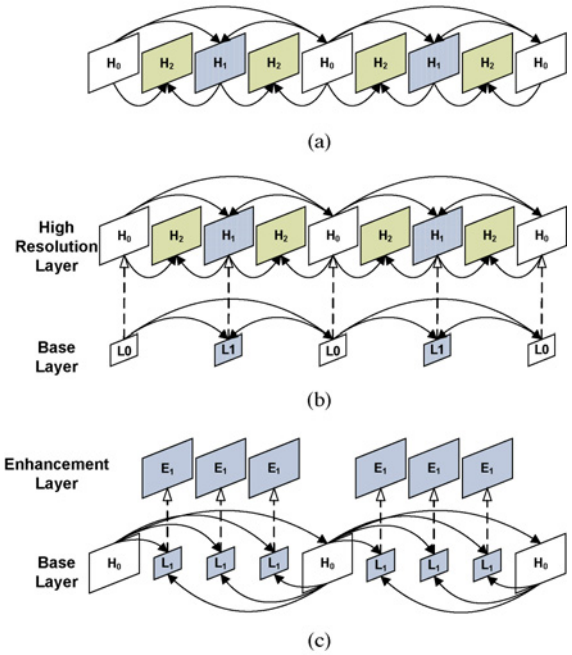

Fig. 2. Three encoding scenarios. (a) Temporal scalable framework. (b) Temporal and spatial scalable framework that can be implemented in H.264/SVC. (c) MR-DVC.

Note that our framework is significantly different from H.264/SVC [35]. The SVC coder has a scalable bit-stream composed of several substreams which are valid bit-streams for some target decoder. It allows temporal, spatial and SNR scalability. However, the SVC scalability does not focus on reduction of encoding complexity. Fig. 2(a) illustrates the frame layer configuration for a temporal scalable codec. The first layer is formed by the H_0 frames, the second layer by the H_0 and H_1 fames, and the third layer by all the frames. For hybrid video codecs, such as H.264/AVC, temporal scalability can be enabled by restricting motion-compensated prediction to reference frames of the same or lower layer. Fig. 2(b) represents a combination of temporal and spatial scalability as can be implemented in H.264/SVC. Note that both high and low resolution layers are conventionally encoded. Motion estimation is performed in both layers. Therefore, the encoding complexity increases proportionally to the number of layers. Nevertheless, each layer is fully decodable except when there is interlayer prediction. Fig. 2(c) represents the spatial scalability of the proposed framework. The encoder only performs motion estimation for the NRWZ-frames and reference frames at the low resolution layer, thus reducing encoding complexity. The high frequency content is WZ coded. Hence, the high resolution layer is actually an enhancement layer for the LR layer and it is not a valid bit-stream on its own nor represents a meaningful video.

A. MR-DVC Encoder Architecture

In general, a frame can be predicted based on multiple reconstructed reference frames in the frame-store, as well as on their corresponding original versions. The syntax element object for reference frames include motion/mode information used for Direct-B prediction for B-frames, and generation of motion vector predictors for fast motion estimation.

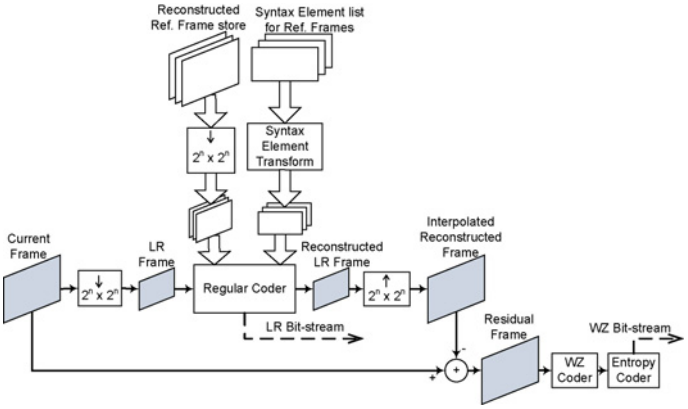

Fig. 3. Architecture for the NRWZ coding mode.

In the NRWZ coding mode, shown in Fig. 3, all the frames in the reference lists and the current frame are decimated by a factor of $2^n \times 2^n$, where n can be selected based on a complexity reduction target. Then, the low resolution current frame is encoded, generating a reduction in the computational complexity and creating the LR layer bit-stream. The quantization parameter used is the same as that corresponding to the target quality for the frame. Note that the syntax element object list for reference frames is also transformed into an appropriate form for reduced resolution encoding. This operation consists in reducing the resolution of the motion vector field and/or mode decisions for the reference frames. This enables not only direct-B prediction for B-frames but also fast motion estimation at reduced resolution. The decoder must duplicate the same process to decode the direct-B modes appropriately. In order to create the enhancement layer, i.e., the WZ layer, the encoder computes the difference between the full resolution original frame and the interpolated reconstruction of the LR coded frame, denoted the Laplacian residue. This residual frame is sent to the decoder using a WZ coder. It is easy to see that the encoder complexity for the NRWZ frame is reduced proportionally to the decimation factor (with proper overhead due to decimation, interpolation, and WZ coding operations).

B. MR-DVC Decoder Architecture

The decoder architecture is presented in Fig. 4. First, the LR image is decoded and interpolated with the same interpolator used at the encoder, generating the decoded base layer. Note that even though decoding only the base layer generates big PSNR variations between frames, it does allow for real-time decoding on power-constrained devices without drop in frame rate, something that is not feasible in traditional temporal Wyner–Ziv coding.

The optional process of enhancement begins with the generation of the SI that will be used to decode the WZ information. In the enhancement process, the decoded interpolated frame and the reference frames are used to create what we call the semi super-resolution (SSR) version of the current frame [22]. Except for the current frame, the other frames used are already at high resolution. Note that the performance of

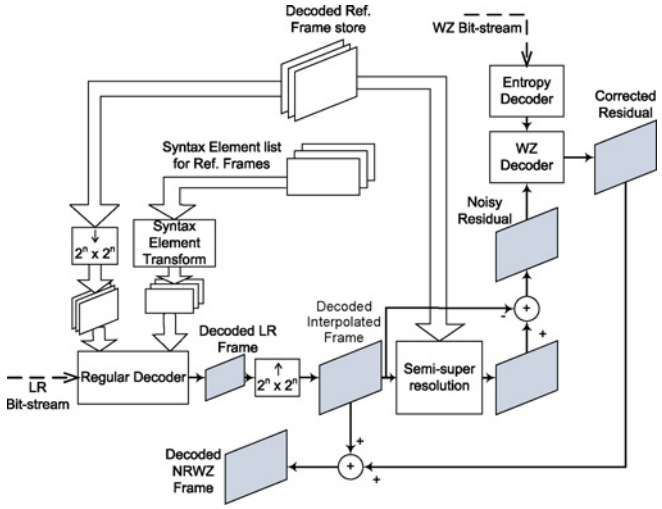

Fig. 4. Architecture for the NRWZ decoder.

any WZ encoder is heavily dependent on the efficiency of the generation of the SI. Once the SSR frame is generated, the interpolated LR decoded frame is subtracted from it. The resulting residual frame is the actual SI frame to be used in the WZ decoder. The WZ decoder decodes the WZ bit-stream layer with the SI residual frame acting as a noisy version of the original transmitted residual frame. The decoded residual frame is finally added to the interpolated LR frame to obtain the final decoded frame.

C. Wyner–Ziv Coder

The Wyner–Ziv codec used here works in the block transform domain on the Laplacian residue frame. In a codec where multiple transforms can be used, for example, AVC fidelity range extensions [36], the largest transform size is preferred but any one can be used. The model assumes that the block transform coefficients, denoted by random variable X, are Laplacian distributed with standard deviation σ_X. The corresponding coefficient in the SI frame is denoted Y. After the block-transform of the residual frame is computed, the coefficients X are quantized with a uniform deadzone quantizer. The random variable that represent the quantized coefficients is denoted by Q, which takes values from the set $\Omega_Q = \{-q_{\max}, -q_{\max+1}, ..., -1, 0, 1, ..., q_{\max-1}, q_{\max}\}$. Cosets are next computed on Q with modulus M to obtain the coset random variable C as follows:

$$C = \psi(Q, M) = \begin{cases} Q - M\lfloor Q/M \rfloor, & Q - M\lfloor Q/M \rfloor < M/2 \\ Q - M\lfloor Q/M \rfloor - M, & Q - M\lfloor Q/M \rfloor \geq M/2. \end{cases} \quad (1)$$

The values of C are taken from the set $\Omega_C = \{\lfloor -(M-1)/2 \rfloor, ..., -1, 0, 1, ..., \lfloor (M-1)/2 \rfloor\}$. In the implemented reversed-complexity coding mode, the parameters QP and M are different for each frequency (i, j) of coefficient x_{ij}, and can be varied from macroblock-to-macroblock. The QP and M coding parameters are chosen based on an estimate of the noise statistics between the SI block and the original one, based on a chosen model (see Section IV). In this paper, a combination of the number of bits spent to code the corresponding residual block in the LR layer and an edge activity measure in the coded block is used to estimate the model parameters (see Section V). Also, only some low-to-mid frequency coefficients are mapped to cosets and sent for each block, while the rest are forced to 0. The number of coefficients transmitted in zigzag scan order is determined based on the noise level estimation. The DC coefficient is sent without coset computation. More details about the parameter selection mechanism are provided in Section IV.

If the quantization bin q corresponds to interval $[x_l(q), x_h(q)]$, then the probability of the bin $q \in \Omega_Q$, and the probability of a coset index $c \in \Omega_C$ are given by the probability mass functions

$$p(q) = \int_{x_l(q)}^{x_h(q)} f_X(x)dx \quad (2)$$

$$p(c) = \sum_{q \in \Omega_Q, \psi(Q,M)=c} p(q) = \sum_{q \in \Omega_Q, \psi(Q,M)=c} \int_{x_l(q)}^{x_h(q)} f_X(x)dx \quad (3)$$

where $f_X(x)$ is the pdf of X. Because the distribution $p(c)$ is symmetric for odd M, has 0 as its mode, and decays with increasing magnitude, the entropy coder for Q that already exists in the conventional codec can be reused for C. We note however that a special entropy coder designed specifically for coset indices should be more efficient. While in this paper, we reuse the same entropy coder as in H.264/AVC in order to minimize the modifications needed to the conventional codec, in the future we will investigate special entropy coding for block cosets.

If Y corresponds to the unquantized SI available only for decoding, then at the decoder, the minimum MSE reconstruction function $\hat{X}_{YC}(y, c)$ based on SI y and received coset index c, is given by

$$\hat{X}_{YC}(y, c) = E(X|Y = y, C = c) = \frac{\sum_{q \in \Omega_Q, \psi(Q,M)=c} \int_{x_l(q)}^{x_h(q)} x f_{X|Y}(x, y)dx}{\sum_{q \in \Omega_Q, \psi(Q,M)=c} \int_{x_l(q)}^{x_h(q)} f_{X|Y}(x, y)dx}. \quad (4)$$

In the decoder, the same estimate for the model parameters based on bit rate at the LR layer and edge activity, is obtained. These model parameters not only yield the same QP and M parameters applied during encoding, but are also used to obtain the optimal reconstruction based on (4). Because the exact computation of (4) is difficult, approximations and interpolations on various precomputed tables can be useful.

III. Semi Super-Resolution Frame Generation and WZ Decoding

Our method, which is an improvement on [24], iteratively computes the SSR frame followed by WZ decoding in multiple passes. A block diagram of the process is shown in Fig. 5.

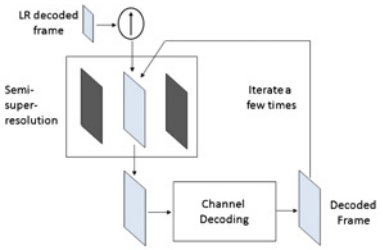

Fig. 5. SI generation. Iterate between SI generation and WZ decoding.

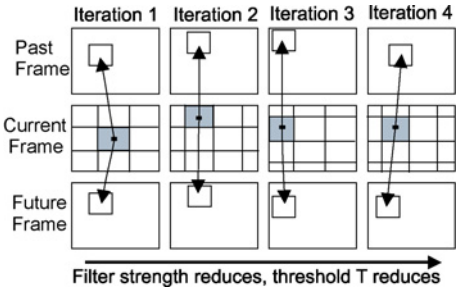

Fig. 6. Semi super-resolution for nonreference WZ frames. Filtered frames are used for motion estimation, but unfiltered frames are used for motion compensation.

Let the interpolated LR reconstructed frame be F_0. Let $SS(F, FS)$ denote the SSR operation to yield a higher resolution version F^{HR} of F based on the stored frames FS. Also, let $D_{WZ}(RF, b_{WZ})$ denote the WZ decoding operation yielding a corrected version of the residual frame based on our noisy version RF using the WZ layer bit-stream b_{WZ}. Then, iterative decoding comprises the following steps for $i = 0, 1, 2, \ldots, N-1$:

$$F_i^{HR} = SS(F_i, FS) \tag{5}$$

$$F_{i+1} = D_{WZ}(F_i^{HR} - F_0, b_{WZ}) + F_0. \tag{6}$$

In the first iteration, similar to an example-based algorithm [37], we seek to restore the high frequency information of an interpolated block through searching in previously decoded key frames for a similar block, and by adding the high frequency of the chosen block to the interpolated one.

The past and future reference frames in the frame-store are low pass filtered. The low pass filter is implemented through down-sampling followed by an up-sampling process (using the same decimator and interpolator applied to the nonkey frames). The high frequency of the reference frames is the residue between the original frame and its filtered version. If F denotes a frame then $F = L + H$, where L is the decimated and interpolated (filtered) version of F, while H is the residue, or its high frequency.

A block-matching algorithm as described below, is applied on the current decoded frame F_i, for $i = 0, 1, 2, \ldots, N-1$, to obtain the SSR frame F_i^{HR}. For every 8×8 block in F_i, the best subpixel motion vectors in the past and future frames are computed to minimize the sum of absolute differences (SAD) between it and low pass filtered versions of past and future frames, respectively. If the best low pass predictor blocks are denoted as L_p and L_f in the past and future filtered frames respectively, several candidate predictors are calculated as $\alpha L_p + (1-\alpha)L_f$. The mixing factor α assumes values between 0 and 1, and in our implementation we used $\alpha \in \{0.0, 0.25, 0.5, 0.75, 1.0\}$. The best mixing factor α^* is chosen to be the one that yields the least SAD between the current block and its predictor.

Next, if the SAD of the best predictor computed for a block is lower than a threshold T, the block is updated in the SSR frame F_i^{HR} as follows. Let H_p and H_f denote the high-frequency components of the best predictor blocks in the past and future frames, respectively. Then in the first iteration, $\alpha^* H_p + (1-\alpha^*)H_f$ is added to the corresponding block in the decoded interpolated frame F_0 to obtain the corresponding block in the super-resolved version F_0^{HR}. In subsequent iterations, the block in the current decoded frame F_i is completely replaced by the unfiltered version of its best predictor: $\alpha^*(H_p + L_p) + (1-\alpha^*)(H_f + L_f)$ to obtain the corresponding block in F_i^{HR}. If the best SAD is larger than a threshold T, then the block in F_i^{HR} is just copied from the corresponding block in F_i.

During the first iteration when only high-frequencies are added, we need to be careful not to add spurious noise in cases where no close matches are found with the low pass-based matching operation. Hence, we use a confidence factor based on the best SAD to scale the high frequency component before adding to the decoded interpolated block. We assume that the lower the SAD, the higher the confidence we have, and thus the more high frequency information is added.

From iteration to iteration three things are changed: 1) the strength of the low pass filter used prior to block-matching is gradually reduced; 2) the grid of the block matching is offset; and 3) the threshold T is gradually reduced. In practice, the low pass filtering operation for the reference frames prior to block-matching is eliminated after one or two iterations as the frame becomes more and more accurate. The grid for block matching is offset at each iteration to smooth out the blockiness and to add spatial coherence. For example, for 8×8 blocks, the shifts used in four passes can be (0, 0), (4, 0), (0, 4) and (4, 4). The value for the threshold was empirically found based on performance over a limited training video sequences and then used for all test sequences in the simulations. In our implementation, we used $T_i = \{500, 80, 60, 20, 5\}$ for 8×8 blocks, where i indicates the number of the iteration. This process is illustrated in Fig. 6, not necessarily to any particular offset. In [38] comparisons between different SI generators including the proposed iterative method are presented

IV. Choosing Coding Parameters

In order to make an optimal choice of the quantization and modulus parameters $\{QP, M\}$, we assume a general enough statistical model: $Y = \rho X + Z$ [16], [39], where X is a Laplacian distributed transform coefficient with S.D. σ_X, Z is additive Gaussian noise uncorrelated with X with S.D. σ_Z, and $0 < \rho < 1$ is an attenuation factor expected to decay at higher frequencies. While this is a generalization of the simpler model: $Y = X + Z$ considered in [22], [23], rewriting

as $Y/\rho = X + Z/\rho$ the same known estimation procedure [23], [40] can be applied, simply replacing σ_Z^2 with $(\sigma_Z/\rho)^2$ and Y with Y/ρ during decoding. In the rest of this section, we review the optimal parameter choice mechanism for the $Y = X + Z$ model, emphasizing that in order to use it for the $Y = \rho X + Z$ model, σ_Z needs to be replaced with (σ_Z/ρ), and Y by Y/ρ.

A. Memoryless Coset Codes Followed by Minimum MSE Reconstruction With SI

The first step is to obtain expressions for expected rate and distortion functions for the memoryless coset codes described in Section II, for a given $\{QP, M\}$ pair. Let R_{YC} be the rate assuming an ideal entropy coder for the coset indices, and let D_{YC} be the distortion given SI y and coset index c. It can be shown (see Appendix Section A) that their expected values are given by

$$\begin{aligned} E(R_{YC}) &= -\sum_{c\in\Omega_C}\left\{\sum_{q\in\Omega_Q:\psi(Q,M)=c}\left[m_X^{(0)}(x_h(q)) - m_X^{(0)}(x_l(q))\right]\right\} \\ &\quad\times \log_2\left\{\sum_{q\in\Omega_Q:\psi(Q,M)=c}\left[m_X^{(0)}(x_h(q)) - m_X^{(0)}(x_l(q))\right]\right\} \end{aligned} \tag{7}$$

$$\begin{aligned} E(D_{YC}) &= \sigma_x^2 - \int_{-\infty}^{\infty}\left\{\sum_{c\in\Omega_C} \times \frac{\left(\sum_{q\in\Omega_Q:\psi(Q,M)=c}[m_{X|Y}^{(1)}(x_h(q), y) - m_{X|Y}^{(1)}(x_l(q), y)]\right)^2}{\left(\sum_{q\in\Omega_Q:\psi(Q,M)=c}[m_{X|Y}^{(0)}(x_h(q), y) - m_{X|Y}^{(0)}(x_l(q), y)]\right)}\right\} \\ &\quad\times f_Y(y)dy \end{aligned} \tag{8}$$

where we defined $m_X^{(i)}(x) = \int_{-\infty}^{x} v^i f_X(v)dv$ and $m_{X|Y}^{(i)}(x, y) = \int_{-\infty}^{x} v^i f_{X|Y}(v, y)dv$.

A viable coding choice is to just use zero-rate coding, where no information is transmitted (i.e., $QP \to \infty$ or $M = 1$). Then the rate is 0 and it can be shown (see Appendix Section B), that the expected distortion based on optimal reconstruction using Y alone is given by

$$\begin{aligned} E(D_Y) &= \sigma_X^2 - \int_{-\infty}^{\infty}\left(\int_{-\infty}^{\infty} x f_{X|Y}(x, y)dx\right)^2 f_Y(y)dy \\ &= \sigma_X^2 - \int_{-\infty}^{\infty} m_{X|Y}^{(1)}(\infty, y)^2 f_Y(y)dy. \end{aligned} \tag{9}$$

B. Laplacian Source With Additive Gaussian Noise

Using the model of a Laplacian source, X, with additive Gaussian noise, Z, we have

$$f_X(x) = \frac{1}{\sqrt{2}\sigma_x} e^{-\left|\frac{x\sqrt{2}}{\sigma_X}\right|} \tag{10}$$

$$f_Z(z) = \frac{1}{\sqrt{2\pi}\sigma_z} e^{-\frac{1}{2}\left|\frac{z}{\sigma_z}\right|^2}. \tag{11}$$

Defining $\beta(x) = \exp[(\sqrt{2}x)/(\sigma_x)]$, we have

$$m_X^{(0)}(x) = \begin{cases} \dfrac{\beta(x)}{2}, & x \le 0 \\ 1 - \dfrac{1}{2\beta(x)}, & x > 0 \end{cases}$$

$$m_X^{(1)}(x) = \begin{cases} \dfrac{\beta(x)}{2\sqrt{2}}(\sqrt{2}x - \sigma_x), & x \le 0 \\ -\dfrac{1}{2\sqrt{2}\beta(x)}(\sqrt{2}x + \sigma_x), & x > 0 \end{cases} \tag{12}$$

The above moments will be used in the calculation of $E(R_{YC})$ in (7). Further defining

$$\gamma_1(x) = erf\left(\frac{\sigma_x x - \sqrt{2}\sigma_z^2}{\sqrt{2}\sigma_x\sigma_z}\right)$$

$$\gamma_2(x) = erf\left(\frac{\sigma_x x + \sqrt{2}\sigma_z^2}{\sqrt{2}\sigma_x\sigma_z}\right) \tag{13}$$

and

$$erf(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}dt \tag{14}$$

then, since $Y = X + Z$, we have

$$\begin{aligned} f_Y(y) &= \int_{-\infty}^{\infty} f_{XY}(x, y)dx \\ &= \frac{1}{2\sqrt{2}\beta(y)\sigma_x} e^{\frac{\sigma_x^2}{\sigma_z^2}}[\gamma_1(y) + 1.0 - \beta(y)^2(\gamma_2(y) - 1.0)] \end{aligned} \tag{15}$$

$$\begin{aligned} f_{X|Y}(x, y) &= \frac{f_{XY}(x, y)}{f_Y(y)} \\ &= \frac{\sqrt{2}\beta(y)}{\sqrt{\pi}\sigma_z} \frac{e^{-\left|\frac{x\sqrt{2}}{\sigma_x}\right| - \frac{1}{2}(\frac{y-x}{\sigma_z})^2 - \frac{\sigma_x^2}{\sigma_z^2}}}{[\gamma_1(y) + 1.0 - \beta(y)^2(\gamma_2(y) - 1.0)]}. \end{aligned} \tag{16}$$

Given $f_{X|Y}(x, y)$, the moments $m_{X|Y}^{(i)}(x, y)$ can now be calculated and used for computing $E(D_{YC})$ in (8) by numerical integration. The $erf()$ function can be evaluated based on a $9th$ order polynomial approximation [41]. All the expected rate and distortion values for a memoryless coset code followed by minimum MSE reconstruction can be evaluated based on these results in conjunction with numerical integration of $f_Y(y)$.

C. Optimal Parameter Choice and Model Prediction

The above results can be used to compute the RD points for the set of all allowable $\{QP, M\}$ combinations, using (7) and (8) given the source and correlation statistics σ_X and σ_Z. In Fig. 7, we present the RD points so obtained for a deadzone quantizer, for the specific case of $\sigma_X = 1$ and $\sigma_X = 0.4$. In Fig. 7(a), each RD curve depicted as "*Const M RD Points*" is generated by fixing M and changing QP at finely sampled intervals of 0.05. However, the following discussion assumes

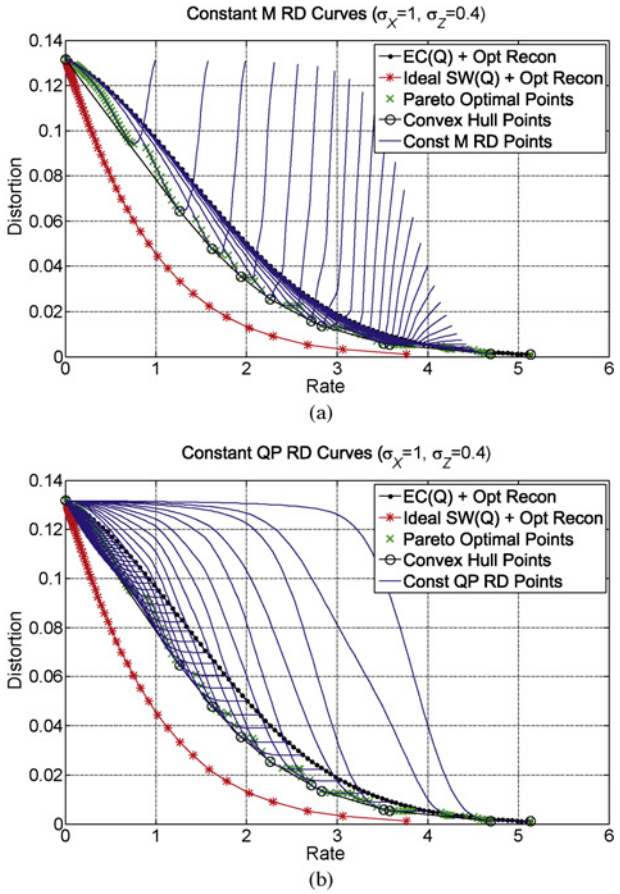

Fig. 7. *RD* curves obtained by varying QP and M for $\sigma_X = 1$ and $\sigma_Z = 0.4$, given by equation along with the corresponding Pareto-optimal and the convex hull set of points. (a) *RD* curves depicted for constant M and varying QP. (b) *RD* curves depicted for constant QP and varying M.

QP to be continuous. The case when $QP \to \infty$ for any M corresponds to the zero-rate case, and yields the *RD* point $\{0, E(D_Y)\}$ where all the curves start, with $E(D_Y)$ given by (9). Alternatively, this point can also be viewed as the $M = 1$ curve that degenerates to a point. Fig. 7(b) shows exactly the same results using constant QP curves and changing M, starting from 1 upward. As $M \to \infty$, the coder becomes the same as a regular encoder not using cosets but still using Optimal Reconstruction based on SI Y. In the figure, the curve for regular encoding Q using Optimal Reconstruction is depicted as "$EC(Q)+Opt\ recon$," where EC indicates entropy coded (equations can be found in the Appendix section D). The curve corresponding to ideal Slepian–Wolf coding of the quantization indices (Q) followed by optimal reconstruction is also shown ("$SW(Q)+Opt\ recon$"), for details, see Appendix Section C. Note that this curve is also the lower convex hull of the *RD* points obtained by ideal Slepian–Wolf coding of the cosets C for all $\{QP, M\}$ combinations, and represents the theoretical upper-bound performance of this framework.

From figure, it is obvious that not all choices for QP and M are necessarily good codes, since they may not outperform regular coding followed by Optimum reconstruction based on Y. However, as it can also be seen from the figure, if $\{QP, M\}$ are correctly chosen, coset coding can outperform regular entropy coding of Q with Optimal Reconstruction. The suboptimal choices for the $\{QP, M\}$ combination can be pruned out by finding the Pareto-optimal set, wherein each point is such that no other point is superior to it, i.e., yields a lower or equal distortion at a lower or equal rate. These points are marked as "$\times$" in Fig. 7(a) and (b). One strategy for parameter choice is to choose the closest Pareto-optimal point to a given target distortion D_t. However, the strategy that yields superior *RD* performance is to operate on the lower convex hull of the set of points generated by all $\{QP, M\}$ combinations. This set of points, called the convex hull set, is a subset of the points in the Pareto-optimal set and is generally much sparser. These points are marked as "$\circ$" in Fig. 7(a) and (b). In order to accurately match the distortion to a given target D_t, ideally the samples should be multiplexed in the right proportion between two nearest codes from the convex hull set. In this paper, however, we simply chose the closest Pareto-optimal point. Note, that there is still room for improvement since there is a gap between coset coding and ideal Slepian–Wolf coding. This is expected because the ideal Slepian–Wolf coder is working at the conditional entropy of $H(Q|Y)$ while the coset coding curves are based on the entropy rate of $H(C)$. However, since we are considering the no-feedback-channel case, it is expected that the Slepian–Wolf bound would not be reached.

D. Distortion Target Matching

The goal of the distortion-matched parameter choice process can now be expressed in terms of (7) and (8). If QP_t is the target quantization step-size of regular encoding, used for key frames, we note that it is advantageous, in our framework, to specify the target distortion D_t in terms of QP_t. The expected distortion from regular encoding followed by MSE reconstruction without SI is given by (see the Appendix for details)

$$\begin{aligned} E(D_Q) &= \sigma_X^2 - \sum_{q\in\Omega_Q} \frac{\left(\int_{xl(q)}^{x_h(q)} x f_X(x)dx\right)^2}{\left(\int_{xl(q)}^{x_h(q)} f_X(x)dx\right)} \\ &= \sigma_X^2 - \sum_{q\in\Omega_Q} \frac{\left(m_X^{(1)}(x_h(q)) - m_X^{(1)}(x_l(q))\right)^2}{\left(m_X^{(0)}(x_h(q)) - m_X^{(0)}(x_l(q))\right)}. \end{aligned} \quad (17)$$

From (17), we obtain D_t for a given QP_t. Thereafter, we search for the optimal code with distortion closest to D_t, but not exceeding it. In practice, this mapping from QP_t to $\{QP, M\}$ can be precomputed and stored in a normalized table for a given σ_Z and $\sigma_X = 1$, for a range of QP_t values in small incremental steps. The QP values are also typically rounded to the same precision as QP_t. To use it for an arbitrary σ_X, the values of QP_t and QP in a normalized table need to be appropriately scaled before and after table-lookup. A limited set of such normalized tables can be stored in a codec for a range of σ_Z values at small increments. Further details about how to generate these tables can be found elsewhere [23], [40].

Reverting back to our $Y = \rho X + Z$ model, the following look-up procedure based on a set of normalized tables is used to obtain the $\{QP^*, M\}$ combination corresponding to a given

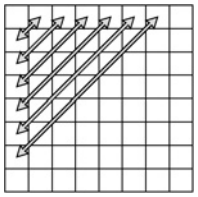

Fig. 8. First seven frequency bands in an 8×8 block of coefficients.

QP_t^*, if the model parameters $\{\rho, \sigma_X, \sigma_Z\}$ are known. We simply have to 1) evaluate $\sigma_Z/(\rho.\sigma_X)$ to find the normalized look-up table to consult from a set; 2) find the closest entry in it corresponding to target $QP_t = QP_t^*/\sigma_X$; 3) read off QP and M; and 4) scale QP to obtain the final $QP^* = QP \times \sigma_X$. The set of allowable QP values will depend on the conventional codec used (H.264 in our implementation).

V. Correlation Statistics Estimation

In this section, we propose a mechanism to estimate the parameters $\{\rho, \sigma_X, \sigma_Z\}$ in the encoder and decoder for our $Y = \rho X + Z$ model within the proposed spatial scalability framework [16], [39]. The estimated parameters will be used to obtain the coding parameters $\{Qp, M\}$ as described in the previous section.

The model parameters are estimated per frequency band (FB) within a block, where an FB is defined as a diagonal in a transform block as illustrated in Fig. 8. Note that the correlation statistics between a coefficient X and the corresponding SI coefficient Y is obviously dependent on the target quantization step-size QP_t for the reference frames and the LR layer of Wyner–Ziv frames. Besides, other vital information such as edge activity measure (see Section V-A) and normalized LR layer rate (see Section V-C) can be conveniently extracted from the LR layer to direct the estimation process. Since any data from the LR layer is available to both decoder and encoder, no overhead bits are needed to convey this information. An alternative approach might explicitly transmit some statistical information, but in this paper we adopt a no-overheads approach.

In the following, we describe the specifics of the estimation models used for estimating the parameters $\{\rho, \sigma_X, \sigma_Z\}$. These models are trained based on (X, Y) pair training data collected from a set of training video sequences for each FB and QP_t, by running the base layer encoder and SSR processing at the decoder for different values of QP_t in small increments.

A. Estimation of σ_X^2-Variance of Laplacian Residual Coefficients

The variance of a Laplacian residual coefficient (σ_X^2) varies from block to block within a frame. It not only depends on QP_t and FB, but also on the high frequency content within the block. If the original frame has a high edge content it is likely that the error between the decoded interpolated version and the original one would be larger. Even though the exact high frequency content in an original frame is not available at the decoder, an edge activity measure, denoted E, computed on the reconstructed LR block can be used as an indicative parameter to estimate σ_X^2. E is computed as the accumulated sum of the absolute difference between adjacent pixels along lines and columns of a macroblock in the decoded interpolated LR frame. It is intuitive to think of the energy at the high frequencies of the Laplacian residual as correlated with E, whereas the energy of the low frequencies in the Laplacian residual is more related to quantization noise introduced by QP_t. In general, the estimated σ_X^2 can be modeled as a function of QP_t, FB and E. That is

$$\sigma_X^2 = f_1(QP_t, FB, E). \tag{18}$$

We next assume σ_X^2 to be proportional to QP_t^2. Further, analysis of the training data reveals that that it suffices to linearly model the variation of σ_X^2 with E for each FB, so that

$$\sigma_X^2 = (k_{1,FB}E + k_{2,FB})QP_t^2 \tag{19}$$

where $k_{i,FB}$ are constants that vary for each frequency band. These parameters are estimated by training based on the training data.

B. Estimation of the correlation parameter

In order to estimate ρ, we use a simplified model assuming that it only depends on QP_t and FB

$$\rho = f_2(QP_t, FB). \tag{20}$$

If TS_{FB,QP_t} represents the training data set for a given QP_t and FB, we estimate

$$\rho_{FB,QP_t} = \text{argmin}_\rho \sum_{(X,Y)\in TS_{FB,QP_t}} \left(\|Y - \rho X\|^2\right). \tag{21}$$

Note that the higher the QP_t, the less the correlation between X and Y. The values of ρ for each QP_t and FB obtained from the training data set can be stored as precalculated tables at both encoder and decoder.

C. Estimation of the Variance of the Gaussian Noise

We conjecture that σ_Z^2 for a macroblock in the enhancement layer is well indicated by the residual error rate R used to code a co-located 8×8 block in the LR base layer. A higher rate in the LR base layer indicates greater inaccuracy of motion estimation at reduced resolution, and therefore the multiframe super-resolution process is also expected to yield a less accurate estimate of the high-resolution frame at the decoder, leading to an increase in σ_Z^2. Besides, we also assume σ_Z^2 to depend on QP_t, FB and E. Now, we can model σ_Z^2 as

$$\sigma_Z^2 = f_3(QP_t, FB, E, R). \tag{22}$$

Since R above depends on QP_t, we can use normalized rate $R_n = R \times QP_t^2$ in order to remove the effect of QP_t. We also assume σ_Z^2 to be proportional to σ_X^2 for a given FB and R_n, and the effect of QP_t and E to be subsumed within σ_X^2. Further, the variation of σ_Z^2 with R_n is linearly modeled for each FB, so that the estimation model is simplified to

$$\sigma_Z^2 = (k_{3,FB}R_n + k_{4,FB})\sigma_X^2. \tag{23}$$

TABLE I
ENCODING TIME (S) COMPARISON FOR *Foreman CIF* SEQUENCE

	IpPpP		IBPBP		IBIBI	
	H.264	DVC	H.264	DVC	H.264	DVC
Total time	254.7	162.7	303.1	173.7	156.3	64.2
ME time	198.3	125.9	234.0	133.7	125.0	32.9

TABLE II
DECODER TIME EVOLUTION WITH THE NUMBER OF ITERATIONS

Foreman CIF Sequence (Mode IBPBP)		
One Iteration (s)	Two Iterations (s)	Three Iterations (s)
901.34	1680.42	2342.47

Constants $k_{3,FB}$ and $k_{4,FB}$ are estimated by training based on the training data, where for each (X, Y) training pair, the corresponding Z is computed as $Y - \rho X$.

VI. RESULTS

The described framework and the proposed SI generation method were implemented on the KTA software for extensions to the state-of-the-art standard H.264/AVC [42]. Results for the H.263+ version can be found elsewhere [16], [22]–[24]. For the simulations of the MR-DVC and regular H.264/AVC we used fast motion estimation, along with the CAVLC entropy coder, without RD optimization (RDO) and without rate control. We also used search range of 16, 2 reference frames, and spatial direct mode type for B-frames. The decimation factor used for the NRWZ frames was 2×2 (half resolution). The training for the estimation models for $\{\sigma_X, \sigma_Z, \rho\}$ described in Section V, is carried using 20 frames from each of the CIF sequences: *Silent*, *Foreman*, and *Mobile*. The results presented use all 300 frames for sequences: *Foreman*, *Coastguard*, *Akiyo*, *Mobile*, *Silent*, *Hall monitor*, *Soccer*, and *Mother-and-Daughter*.

In the simulations, it is only necessary to select one QP_t depending on the target distortion and desired rate. This QP_t will be used for the base layer coded at low resolution. Once the values of $\{\sigma_X, \sigma_Z, \rho\}$ have been estimated, the mechanism described in Section IV-C will choose the appropriate $\{QP, M\}$ pair yielding a decoded quality similar to that of a conventional codec coded with QP_t.

In Table I, a comparison of the encoding time between the conventional H.264/AVC and the proposed MR-DVC is presented. All the coding tests were made on an Intel® Pentium® D 915 2.80 GHz Dual Core with Windows® OS. The WZ mode was set to operate in $IbPbP$, $IpPpP$ and $IbIbI$ modes where b frames indicate NRWZ B-like frames at half resolution and p is a disposable NRWZ P-like frame, also at half resolution. For conventional H.264/AVC, p frames refer to disposable P frames [13]. It can be seen that the mixed resolution $IpPpP$ mode reduces the total encoding time by 36% compared with conventional encoding in $IpPpP$ mode. Note that if encoding of an NRWZ p frame at half-resolution is exactly quarter of a disposable p frame for conventional H.264/AVC, the complexity reduction would be 37.5%. This

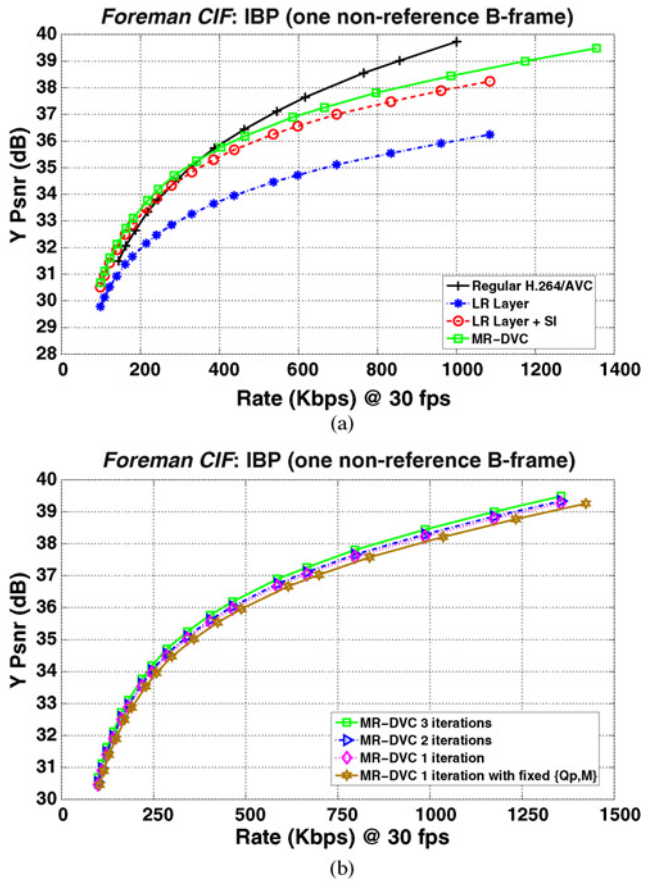

Fig. 9. PSNR results for the Y component of *Foreman CIF* sequence using an $IBPBP$ GOP. (a) We compare conventional H.264; the LR base layer after up-sampling; the key frames along with first SI generated; and the MR-DVC after 3 iterations. (b) We compare the performance of the MR-DVC after 1, 2, and 3 iterations, and with fixed coding parameters.

shows that the overhead in Wyner–Ziv coding is limited. For the $IbPbP$ mode we achieved a 42% reduction in total encoding time over conventional $IBPBP$. Even though NRWZ b frames demand quarter the complexity of conventional B-frames, the overall reduction is higher since a conventional B-frame is more expensive to encode than a P-frame. Even lower encoding complexity can be achieved using intra key frames, e.g., in a $IbIbI$ mode, where a reduction of 59% is achieved over conventional $IBIBI$.

At the decoder, the most complex operation is the semi super-resolution SI generation process. In principle, the complexity of SI generation per iteration is of the same order as that of encoding of a regular B-frame. More iterations proportionally increase the decoding time. In Table II, we present the actual decoding times for multiple iterations of the same (*Foreman*) sequence. Note, however, that these decoding times are quite high since our SI generation implementation is not optimized.

In Fig. 9(a), we show the coding performance of: 1) the reversed-complexity codec (after three iterations) operating in $IbPbP$ mode; 2) conventional H.264/AVC codec operating in $IBPBP$ mode; 3) LR layer that is formed by just up-sampling the NRWZ-frames and the key frames; and 4) the key frames of the LR layer with the NRWZ frames replaced by the SI

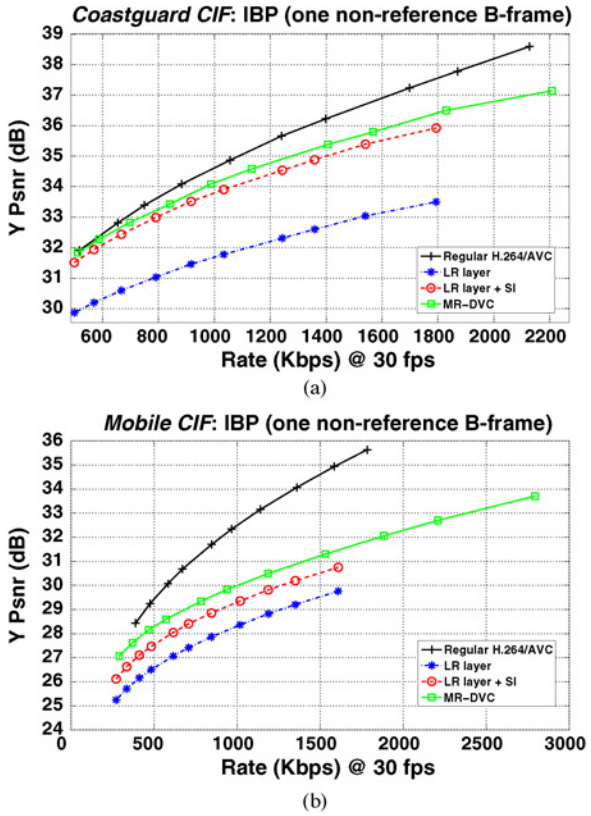

Fig. 10. PSNR results for Y component at $IBPBP$ mode. Comparing: conventional H.264; the LR base layer after up-sampling; the key frames of the base layer along with first SI generated; and the MR-DVC, for (a) *Coastguard CIF*, (b) *Mobile CIF*.

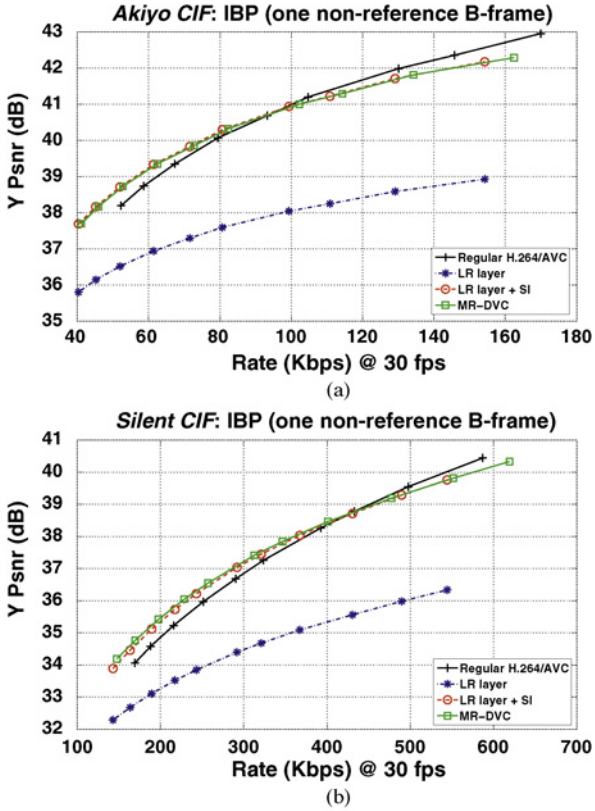

Fig. 11. PSNR results for Y component at $IBPBP$ mode. Comparing: conventional H.264; the LR base layer after up-sampling; the key frames of the base layer along with first SI generated; and the MR-DVC, for (a) *Akiyo CIF*, (b) *Silent CIF*.

generated during the first iteration and without WZ decoding. The PSNR is computed from the luminance component as the arithmetic mean of the PSNR values for each frame [43]. It can be seen that the first SI already outperforms the regular spatial interpolation of the LR layer. For low rates, the key frames with the first SI and the WZ modes outperform conventional coding. At high rates, conventional encoding outperforms WZ coding. However, the WZ layer yields an improvement when compared to the first SI applied to the LR layer. In Fig. 9(b) we show the performance of: 1) the reversed-complexity codec after one, two, and three iterations operating in $IbPbP$ mode; and 2) the reversed-complexity codec without the mechanism for choosing coding parameters (QP is fixed to the same value as the base layer and $M = 15$). The difference between each iteration is around 0.1 dB. For this particular sequence more iterations would not reflect in any more gains. Even though the PSNR difference at each iteration is small the subjective difference is significant. Also, we can see that the reversed-complexity coding mode, with fixed values of $\{QP, M\}$ is outperformed by the same mode using the proposed mechanism for choosing coding parameters.

Additional results for the same $IbPbP$ mode are presented in Figs. 10 and 11 for *Coastguard*, *Mobile*, *Akiyo* and *Silent* sequences. It can be observed that the reversed-complexity WZ mode is competitive. For *Akiyo* and *Silent* sequences the coding mechanism only sends limited information for the WZ layer because it predicts that sending more information is not helpful. That is the reason why there is not a significant gap between the WZ mode and the curve using only the first SI. Our worst results are for the *Mobile CIF* sequence, where there is the largest gap between conventional WZ coding. This is due to the high edge content and because the amounts of nontranslational motion were higher than our SI generation process is able to restore. The conventional H.264/AVC coder has an advantage in this regard since it can adaptively use small block sizes up to 4×4 for motion estimation. Our SI generation process, however, currently works only with 8×8 blocks.

In Fig. 12, we compare the performances of our MR-DVC in $IpPpP$ mode (where p frames are at half resolution) against a conventional H.264 codec in $IpPpP$ mode for *Foreman* and *Mother-and-Daughter* sequences. We find that for the *Foreman CIF* sequence, the gain of the WZ layer is significant when compared to using only SI. It can be observed that, at low rates for many sequences, the SI and especially the WZ modes outperform conventional H.264/AVC in average PSNR, mainly due to accurate SI generation applied to the mixed resolution framework. The MR-DVC scheme can be viewed as an interpolative coding scheme which is known for good performance at low bit-rates. Other interpolative coding schemes have been used in image compression with better performance than regular compression for low rates [44]. One

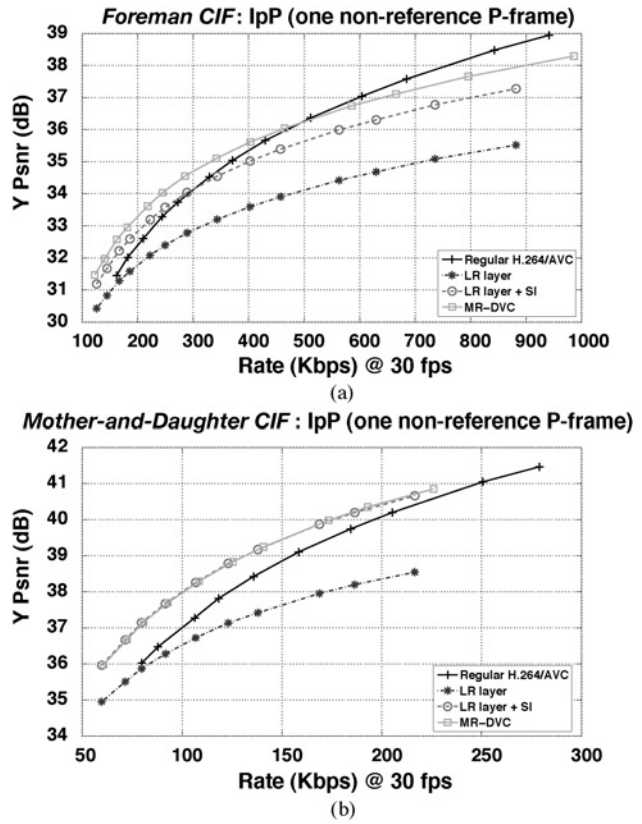

Fig. 12. PSNR results for Y component at $IpPpP$ mode. Comparing conventional H.264 at $IpPpP$ mode, the low resolution base layer, the base layer along with the SI and the MR-DVC. (a) *Foreman*. (b) *Mother-and-Daughter*.

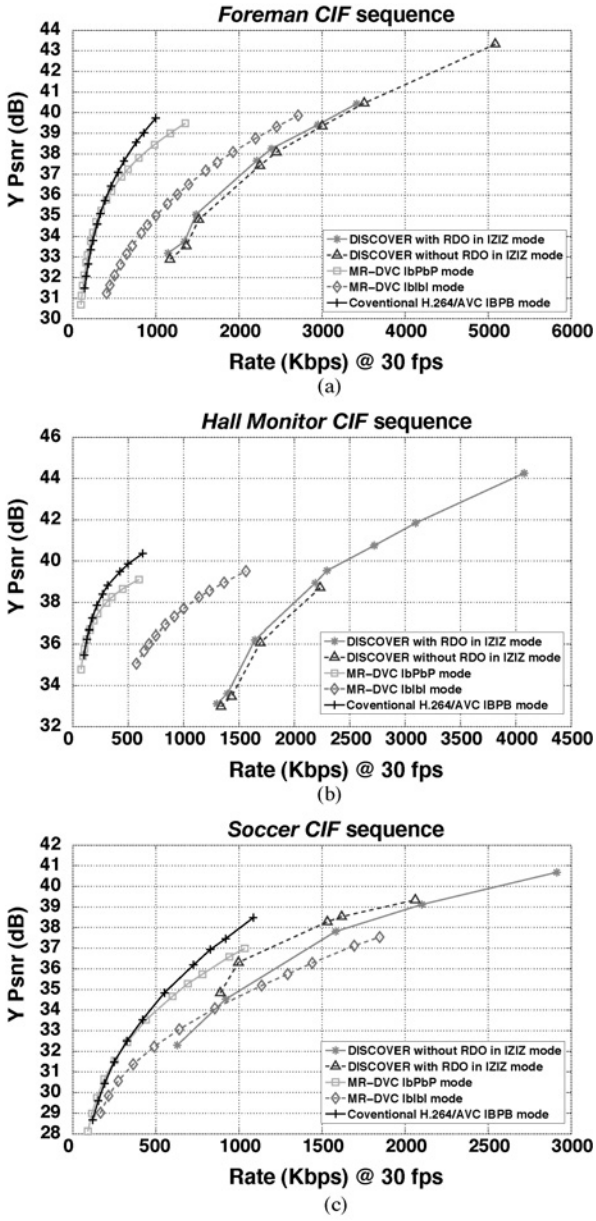

Fig. 13. PSNR results for Y component. Comparing MR-DVC with other WZ architecture. (a) *Foreman CIF*. (b) *Hall Monitor CIF*. (c) *Soccer CIF*.

caveat of our scheme is the potentially high PSNR variation among key and SI frames. Depending on the quality of the SI, the PSNR variation can be as small as 0.3 dB or as large as 6 dB. Generally, for low motion sequences like *Silent*, *Akiyo*, and *Mother-and-Daughter* the maximum PSNR variation is low. For high motion sequences such as *Coastguard* or *Mobile*, the PSNR variation is high.

For a final comparison, in Fig. 13, we compare MR-DVC, working in $IbIbI$ and $IbPbP$ modes, against one of the most popular WZ architectures: the DISCOVER codec [45]. As mentioned before, our WZ mode does not use RD optimization. The simulations for the DISCOVER codec were carried with and without RDO, with fast motion estimation, CAVLC entropy coder, and working in $IZIZI$ mode where Z means a WZ frame coded entirely by a Wyner–Ziv coder. It is also important to note that the DISCOVER codec uses sequence dependent QP tables, not to maximize the RD performance, but to have a uniform subjective quality through the whole sequence. Also, it uses a feedback channel which is not used by the MR-DVC. In terms of complexity, using RDO, the DISCOVER codec takes about 165 seconds to encode the key frames of an entire CIF sequence in average, which makes the encoding complexity comparable to our scheme without RDO in $IbPbP$ mode. However, without RDO, the DISCOVER codec can encode the key frames in approximately 20 seconds. The MR-DVC architecture does not achieve such low encoding complexity as indicated in Table I. In the least-complexity ($IbIbI$) mode, the encoding time was about 64 seconds. Nevertheless, as it can be seen in Fig. 13, the MR-DVC achieves better RD performance for *Foreman* and *Hall Monitor CIF* sequences. The *Hall Monitor* sequence can be consider a low motion sequence where inter coding is very efficient. SI generation process using temporal frame interpolation, as in the DISCOVER, can achieve good results for low motion sequences. However, our scheme can generate a more accurate SI since the high frequency components can be recovered effectively from the key frames. For the *Soccer CIF* sequence DISCOVER outperformed our MR-DVC only in $IbIbI$ mode, even with a better SI [38], the MR-DVC apparently did not selected correctly the coding parameters in this particular case.

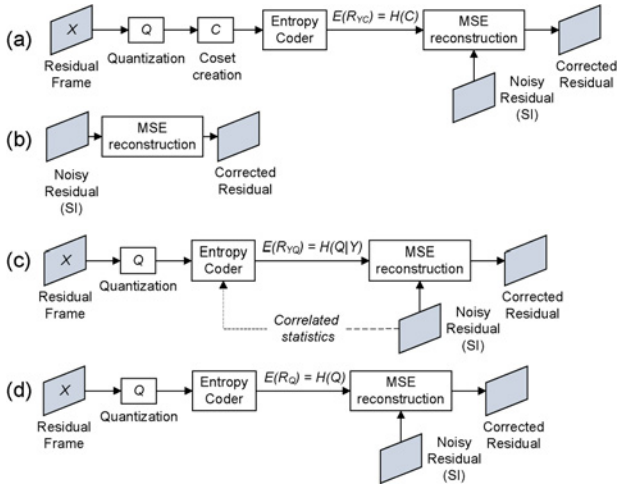

Fig. 14. Different encoding scenarios.

VII. Conclusion

In this paper, we have presented a mixed resolution Wyner–Ziv video coding framework, and proposed a motion-based semi super-resolution SI generation mechanism, and a mechanism for selecting coding parameters based on a statistical correlation estimation. The SI generation and Wyner–Ziv layer decoding are iteratively carried in order to allow for better performance of the Wyner–Ziv decoder. The SI generation process is critical to the efficiency and robustness of the codec. Our Wyner–Ziv coding mode does not use a feedback channel. Hence, a correlation estimation method was proposed in order to choose the right coding parameters. The results show that MR-DVC is competitive and allows reduction of encoding complexity, irrespective of the implementation of the core encoder. It may even outperform regular coding for slow motion sequences at low bit rates. The work in this coder is evergreen. We are continuously working on improving the SI generation process. Improved channel codes and entropy codes for the WZ layer are also being explored.

Appendix
R-D Characterization

A. Memoryless Coset Codes Followed by Minimum MSE Reconstruction

Assuming an ideal entropy coder for the coset indices as show in Fig. 14(a), the expected rate would be the entropy of the source C, given by

$$E(R_{YC}) = H(C) = -\sum_{c\in\Omega_C} p_c(c)\log_2 p_c(c)$$
$$= -\sum_{c\in\Omega_C}\left\{\sum_{q\in\Omega_Q:\psi(Q,M)=c}\int_{x_l(q)}^{x_h(q)} f_X(x)dx\right\}$$
$$\times\log_2\left\{\sum_{q\in\Omega_Q:\psi(Q,M)=c}\int_{x_l(q)}^{x_h(q)} f_X(x)dx\right\}$$

That can be rewritten as

$$E(R_{YC}) = -\sum_{c\in\Omega_C}\left\{\sum_{q\in\Omega_Q:\psi(Q,M)=c}\left[m_X^{(0)}(x_h(q)) - m_X^{(0)}(x_l(q))\right]\right\}$$
$$\times\log_2\left\{\sum_{q\in\Omega_Q:\psi(Q,M)=c}\left[m_X^{(0)}(x_h(q)) - m_X^{(0)}(x_l(q))\right]\right\}$$

where $m_X^{(i)}(x) = \int_{-\infty}^{x} v^i f_X(v)dv$.

Assuming the minimum MSE reconstruction function in (4), the expected distortion D_{YC}, given y and c, is given by

$$E(D_{YC}|Y=y, C=c) = E([X-\hat{X}_{YC}(y,c)]^2|Y=y, C=c)$$
$$= E(X^2|Y=y, C=c) - \hat{X}_{YC}(y,c)^2$$

where $\hat{X}_{YC}(y,c) = E(X|Y=y, C=c)$. Marginalizing over y and c yields

$$E(D_{YC})$$
$$= E(X^2) - \int_{-\infty}^{\infty}\left\{\sum_{c\in\Omega_C}\hat{X}_{YC}(y,c)^2 p_{C|Y}(C=c|Y=y)\right\} f_Y(y)dy.$$

That can be expressed as

$$E(D_{YC})$$
$$= \sigma_X^2 - \int_{-\infty}^{\infty}\left\{\sum_{c\in\Omega_C}\left(\frac{\sum_{q\in\Omega_Q:\psi(Q,M)=c}\int_{x_l(q)}^{x_h(q)} x f_{X|Y}(x,y)dx}{\sum_{q\in\Omega_Q:\psi(Q,M)=c}\int_{x_l(q)}^{x_h(q)} f_{X|Y}(x,y)dx}\right)^2\right.$$
$$\left.\times\ p_{C|Y}(C=c|Y=y)\right\} f_Y(y)dy$$

where $p_{C|Y}(C=c|Y=y)$ is the conditional probability mass function of C given Y, that is

$$p_{C|Y}(C=c|Y=y) = \sum_{q\in\Omega_Q:\psi(Q,M)=c}\int_{x_l(q)}^{x_h(q)} f_{X|Y}(x,y)dx. \quad (24)$$

Then, we have

$$E(D_{YC}) = \sigma_X^2$$
$$-\int_{-\infty}^{\infty}\left\{\sum_{c\in\Omega_C}\times\frac{\left(\sum_{q\in\Omega_Q:\psi(Q,M)=c}\int_{x_l(q)}^{x_h(q)} x f_{X|Y}(x,y)dx\right)^2}{\left(\sum_{q\in\Omega_Q:\psi(Q,M)=c}\int_{x_l(q)}^{x_h(q)} f_{X|Y}(x,y)dx\right)}\right\}$$
$$\times f_Y(y)dy.$$

Defining

$$m_{X|Y}^{(i)}(x,y) = \int_{-\infty}^{x} v^i f_{X|Y}(v,y)dv \quad (25)$$

we can rewrite $E(D_{YC})$ as

$$E(D_{YC}) = \sigma_X^2 - \int_{-\infty}^{\infty} \left\{ \sum_{c\in\Omega_C} \times \frac{\left(\sum_{q\in\Omega_Q:\psi(Q,M)=c} [m^{(1)}_{X|Y}(x_h(q), y) - m^{(1)}_{X|Y}(x_l(q), y)] \right)^2}{\left(\sum_{q\in\Omega_Q:\psi(Q,M)=c} [m^{(0)}_{X|Y}(x_h(q), y) - m^{(0)}_{X|Y}(x_l(q), y)] \right)} \right\} \times f_Y(y)dy.$$

B. Zero Rate Encoder With Minimum MSE Reconstruction With SI

A viable coding choice is to just use zero-rate coding [Fig. 14(b)], where no information is transmitted (i.e., $QP \to \infty$ or $M = 1$). In this case, the decoder performs the minimum MSE reconstruction function $\hat{X}_Y(y)$

$$\hat{X}_Y(y) = E(X|Y = y) = \int_{-\infty}^{\infty} x f_{X|Y}(x, y)dx = m^{(1)}_{X|Y}(\infty, y)$$

Then, the expected zero-rate-distortion D_Y is given by

$$E(D_Y) = \sigma_X^2 - \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} x f_{X|Y}(x, y)dx \right)^2 f_Y(y)dy = \sigma_X^2 - \int_{-\infty}^{\infty} m^{(1)}_{X|Y}(\infty, y)^2 f_Y(y)dy.$$

C. Ideal Slepian–Wolf Coding Followed by MSE Reconst ruction With SI

Next, we consider the expected rate and distortion when using ideal Slepian–Wolf coding for the quantization bins [Fig. 14(c)]. Note, that the optimal set of point for Slepian–Wolf coding of C will be when $M \to \infty$, i.e., when Slepian–Wolf coding is applied directly on the quantization bins. Then, the ideal Slepian–Wolf coder would use a rate no larger than $H(Q|Y)$. Then, the expected rate is given by

$$E(R_{YQ}) = H(Q|Y) = -\int_{-\infty}^{\infty} \left\{ \sum_{q\in\Omega_Q} p_{Q|Y}(Q = q|Y = y) \log_2 p_{Q|Y}(Q = q|Y = y) \right\} \times f_Y(y)dy.$$

That can be rewritten as

$$E(R_{YQ}) = \int_{-\infty}^{\infty} \left\{ \sum_{q\in\Omega_Q} [m^0_{X|Y}(x_h(q), y) - m^0_{X|Y}(x_l(q), y)] \times \log_2[m^0_{X|Y}(x_h(q), y) - m^0_{X|Y}(x_l(q), y)] \right\} f_Y(y)dy.$$

The expected distortion D_{YQ} is the distortion incurred by a minimum MSE reconstruction function within a quantization bin given the SI y and bin index q. This reconstruction function $\hat{X}_{YQ}(y, q)$ is given by

$$\hat{X}_{YQ}(y, q) = E(X|Y = y, Q = q) = \frac{\int_{x_l(q)}^{x_h(q)} x f_{X|Y}(x, y)dx}{\int_{x_l(q)}^{x_h(q)} f_{X|Y}(x, y)dx}.$$

The above equation can be rewritten as

$$\hat{X}_{YQ}(y, q) = \frac{m^{(1)}_{X|Y}(x_h(q), y) - m^{(1)}_{X|Y}(x_l(q), y)}{m^{(0)}_{X|Y}(x_h(q), y) - m^{(0)}_{X|Y}(x_l(q), y)}.$$

Using this reconstruction, the expected distortion (D_{YQ}) with noise-free quantization bins is given by

$$E(D_{YQ}) = \sigma_X^2 - \int_{-\infty}^{\infty} \left\{ \sum_{q\in\Omega_Q} \frac{\left(\int_{x_l(q)}^{x_h(q)} x f_{X|Y}(x, y)dx \right)^2}{\int_{x_l(q)}^{x_h(q)} f_{X|Y}(x, y)dx} \right\} f_Y(y)dy = \sigma_X^2 - \left[\int_{-\infty}^{\infty} \left\{ \sum_{q\in\Omega_Q} \frac{\left(m^{(1)}_{X|Y}(x_h(q), y) - m^{(1)}_{X|Y}(x_l(q), y) \right)^2}{m^{(0)}_{X|Y}(x_h(q), y) - m^{(0)}_{X|Y}(x_l(q), y)} \right\} \times f_Y(y)dy \right].$$

D. Regular Encoding Followed by Minimum MSE Reconstruction With and Without SI

Now, we consider the rate and distortion if no distributed coding on the quantization bins were done at the encoder [see Fig. 14(d)]. In this case, the expected rate is just the entropy of Q

$$E(R_Q) = H(Q) = -\sum_{q\in\Omega_Q} p_Q(q) \log_2 p_Q(q) = -\sum_{q\in\Omega_Q} \{[m^{(0)}_X(x_h(q)) - m^{(0)}_X(x_l(q))] \times \log_2[m^{(0)}_X(x_h(q)) - m^{(0)}_X(x_l(q))]\}.$$

The decoder can still use distributed decoding if SI Y is available. In this case, the reconstruction function and the corresponding expected distortion are given by the same $\hat{X}_{YQ}(y, q)$ and $E(D_{YQ})$ used for ideal Slepian–Wolf coding followed by minimum MSE reconstruction with SI, respectively. If there is no SI available, the expected distortion D_Q is the distortion incurred by a minimum MSE reconstruction function just based on the bin index q. Such function, $\hat{X}_Q(q)$, is then given by

$$\hat{X}_Q(q) = E(X/Q = q) = \frac{\int_{xl(q)}^{x_h(q)} x f_X(x)dx}{\int_{xl(q)}^{x_h(q)} f_X(x)dx} = \frac{m^{(1)}_X(x_h(q)) - m^{(1)}_X(x_l(q))}{m^{(0)}_X(x_h(q)) - m^{(0)}_X(x_l(q))}$$

and the expected distortion is given by

$$E(D_Q) = \sigma_X^2 - \sum_{q\in\Omega_Q} \frac{\left(\int_{xl(q)}^{x_h(q)} x f_X(x)dx\right)^2}{\left(\int_{xl(q)}^{x_h(q)} f_X(x)dx\right)}$$

$$= \sigma_X^2 - \sum_{q\in\Omega_Q} \frac{\left(m_X^{(1)}(x_h(q)) - m_X^{(1)}(x_l(q))\right)^2}{\left(m_X^{(0)}(x_h(q)) - m_X^{(0)}(x_l(q))\right)}.$$

REFERENCES

[1] J. Slepian and J. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inf. Theory*, vol. 19, no. 4, pp. 471–480, Jul. 1973.

[2] A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Trans. Inf. Theory*, vol. 2, no. 1, pp. 1–10, Jan. 1976.

[3] S. S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (DISCUS): design and construction," in *Proc. IEEE Data Compress. Conf.*, Mar. 1999, pp. 158–167.

[4] A. Aaron, R. Zhang, and B. Girod, "Transform-domain Wyner–Ziv codec for video," in *Proc. SPIE Vis. Commun. Image Process.*, San Jose, CA, vol. 5308. Jan. 2004, pp. 520–528.

[5] R. Puri and K. Ramchandran, "PRISM: A new robust video coding architecture based on distributed compression principles," in *Proc. Allerton Conf. Commun., Control Comput.*, 2002.

[6] Q. Xu and Z. Xiong, "Layered Wyner–Ziv video coding," in *Proc. SPIE Vis. Commun. Image Process.*, San Jose, CA, Jan. 2004, pp. 301–310.

[7] Q. Xu and Z. Xiong, "Layered Wyner–Ziv video coding," *IEEE Trans. Image Process.*, vol. 15, no. 12, pp. 3791–3809, Dec. 2006.

[8] H. Wang, N. M. Cheung, and A. Ortega, "A framework for adaptive scalable video coding using Wyner–Ziv techniques," *EURASIP J. Appl. Signal Process.*, vol. 2006, pp. 1–18, 2006.

[9] M. Tagliasacchi, A. Majumdar, and K. Ramchandram, "A distributed-source-coding based robust spatio-temporal scalable video codec," in *Proc. Pic. Cod. Symp.*, San Francis, CA, Dec. 2004.

[10] X. Wang and M. T. Orchard, "Desing of trellis codes for source coding with side infotmation at the decoder," in *Proc. IEEE Data Compress. Conf.*, 2001, pp. 361–370.

[11] A. Aaron and B. Girod, "Compression with side-information using turbo codes," in *Proc. IEEE Data Compress. Conf.*, 2002, pp. 252–261.

[12] B. Girod, A. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed video coding," *Proc. IEEE*, vol. 93, no. 1, pp. 71–83, Jan. 2005.

[13] T. Weigand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circ. Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003.

[14] E. Peixoto, R. L. de Queiroz, and D. Mukherjee, "Mobile video communications using a Wyner-Ziv transcoder," in *Proc. SPIE Vis. Commun. Image Process.*, San Jose, CA, vol. 14. 2007, pp. 88–91.

[15] A. M. Aaron, S. Rane, R. Zhang, and B. Girod, "Wyner-Ziv coding for video: Applications to compression and error resilience," in *Proc. IEEE Data Compress. Conf.*, 2003, pp. 93–102.

[16] B. Macchiavello, D. Mukherjee, and R. de Queiroz, "A statistical model for a mixed resolution Wyner-Ziv framework," in *Proc. Pic. Cod. Symp.*, Lisboa, Portugal, 2007, pp. 497–500.

[17] A. Aaron, S. Rane, and B. Girod, "Wyner–Ziv video coding with hash-based motion compensation at the receiver," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2004, pp. 3097–3100.

[18] E. Martinian, A. Vetro, J. Yedidia, J. Ascenso, A. Khisti, and D. Malioutov, "Hybrid distributed video coding using SCA codes," in *Proc. IEEE Workshop Multimedia Signal Process. (MMSP)*, Oct. 2006, pp. 258–261.

[19] M. Morbee, J. Prades-Nebot, A. Pizurica, and W. Philips, "Rate-allocation algorithm for pixel-domain distributed video coding without feedback channel," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Honolulu, HI, May 2007, pp. 521–524.

[20] T. Sheng, G. Hua, H. Guo, J. Zhou, and C. W. Chen, "Rate-allocation for transform domain Wyner-Ziv video coding without feedback," in *Proc. 16th ACM Int. Conf. Multimedia*, 2008, pp. 701–704.

[21] C. Yaacoub, J. Farah, and B. Pesquet-Popescu, "Feedback channel suppression in distributed video coding with adaptive rate-allocation and quantization for multiuser applications," *EUSASIP J. Wireless Commun. Netw.*, vol. 2008, p. 13, 2008.

[22] D. Mukherjee, "A robust reversed-complexity Wyner-Ziv video codec introducing sign-modulated codes," HP Lab., Tech. Rep. HPL-2006-80, May 2006.

[23] D. Mukherjee, B. Macchiavello, and R. L. de Queiroz, "A simple reversed-complexity Wyner-Ziv video coding mode based on a spatial reduction framework," in *Proc. SPIE Vis. Commun. Image Process.*, vol. 6508. Jan. 2007, pp. 65081Y1–65081Y12.

[24] B. Macchiavello, R. de Queiroz, and D. Mukherjee, "Motion-based side-inforamtion generation for a scalable Wyner-Ziv video coding," in *Proc. IEEE Int. Conf. Imag. Process.*, San Antonio, TX, 2007, pp. 413–416.

[25] G. Cote, B. Erol, M. Gallant, and F. Kossentini, "H.263+: video coding at low bit-rates," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 7, pp. 849–866, Nov. 1998.

[26] X. Artigas and L. Torres, "Iterative generation of motion-compensated side information for distributed video coding," in *Proc. IEEE Int. Conf. Image Process.*, vol. 1. 2005, pp. 833–836.

[27] J. Ascenso, C. Brites, and F. Pereira, "Motion compensated refinement for low complexity pixel based distributed video coding," in *Proc. IEEE Conf. Adv. Video Signal Based Surveil.*, Sep. 2005, pp. 593–598.

[28] W. Weerakkody, W. Fernando, J. Martinez, P. Cuenca, and F. Quiles, "An iterative refinement technique for side-information generation in DVC," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jul. 2007, pp. 164–167.

[29] D. Mukherjee, "Parameter selection for Wyner-Ziv coding of Laplacian sources," in *Proc. SPIE Vis. Commun. Image Process.*, San Jose, CA, 2007.

[30] M. Wu, G. Hua, and C. W. Chen, "Syndrome-based lightweight video coding for mobile wireless application," in *Proc. Int. Conf. Multimedia Expo.*, 2006, pp. 2013–2016.

[31] Y. Steinberg and N. Merhav, "On successive refinement for the Wyner–Ziv problem," *IEEE Trans. Inf. Theory*, vol. 50, no. 8, pp. 1636–1654, Aug. 2004.

[32] Z. Li and E. J. Delp, "Wyner–Ziv video side estimator: conventional motion search methods revisited," in *Proc. IEEE Int. Conf. Image Process.*, vol. 1. Sep. 2005, pp. 825–828.

[33] Z. Li, L. Liu, and E. J. Delp, "Rate-distortion analysis of motion side estimation in Wyner–Ziv video coding," *IEEE Trans. Image Process.*, vol. 16, no. 1, pp. 98–113, Jan. 2007.

[34] L. W. Kang and C. S. Lu, "Wyner-Ziv video coding with coding mode-aided motion compensation," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2006, pp. 237–240.

[35] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1103–1120, Sep. 2007.

[36] G. Sullivan, P. Topiwala, and A. Luthra, "The H.264/AVC advanced video coding standard: overview and introduction to the fidelity range extensions," presented at the SPIE Conf. Appl. Digital Image Process., San Diego, CA, Aug. 2004.

[37] W. Freeman, T. Jones, and E. Pasztor, "Example-based super-resolution," *IEEE Comput. Graph. Appl.*, vol. 22, no. 2, pp. 56–65, Apr. 2002.

[38] B. Macchiavello, F. Brandi, E. Peixoto, R. L. de Queiroz, and D. Mukherjee, "Side-information generation for temporal and spatial scalable Wyner-Ziv codecs," *EURASIP J. Image Video Cod.*, to be published.

[39] B. Macchiavello, R. de Queiroz, and D. Mukherjee, "Parameter estimation for an AVC-based distributed video coder," in *Proc. IEEE Int. Conf. Image Process.*, San Diego, CA, 2008, pp. 1124–1127.

[40] D. Mukherjee, "Optimal parameter choice for Wyner-Ziv coding of Laplacian sources with decoder side-information," HP Lab., Tech. Rep., HPL-2007-34, 2007.

[41] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes in C*, 2nd ed. Cambridge: U.K., Cambridge Univ. Press, 1992.

[42] K. Sühring, H.264/AVC Software Coordination, *KTA Software* [Online]. Available: http://www.iphome.hhi.de/suehring/tml/

[43] T. Tan, G. Sullivan, and T. Wedi, "Recommended simulation common conditions for coding efficiency experiments," in *Proc. 31st Meet. ITU-T Video Cod. Experts Group ITU-T SG16 Q.16 Doc. VCEG-AE010*, Morocco, Jan. 2007.

[44] B. Zeng and A. N. Venetsanopoulos, "A jpeg-based interpolative image coding scheme," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 5. 1993, pp. 393–396.

[45] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Ouaret, "The DISCOVER codec: Architecture, techniques and evaluation," in *Proc. Pic. Cod. Symp.*, Lisboa, Portugal, vol. 17, no. 9. Nov. 2007, pp. 1103–1120.

Bruno Macchiavello received the Bachelor degree in electronic engineering from Pontifical Catholic University, Lima, Peru, in 2001, and the M.Sc. and D.Sc. degrees in electrical engineering from the Universidade de Brasilia, Brasilia, Brazil, in 2004 and 2009, respectively.

Currently, he is with the Department of Computer Science, Unviersidade de Brasilia. His main research interests include video coding, image processing, and distributed source coding.

Debargha Mukherjee (M'99–SM'09) was born in Kolkata, India. He received the B.Tech. degree from the Indian Institute of Technology, Kharagpur, India, in 1993, and the M.S. and Ph.D. degrees in electronics and communication engineering from the University of California, Santa Barbara, in 1995 and 1999, respectively.

Since 1999, he has been with Hewlett-Packard Laboratories, Palo Alto, CA, where he is currently a Senior Research Scientist in the Multimedia Communications and Networking Lab. Between 2002 and 2004, he was involved with the MPEG-21 standardization activities and several of his proposals were adopted in MPEG-21 Part 7 on Digital Item Adaptation. He is the coauthor of more than 50 papers in refereed conferences and journals. His research interests include standard and distributed source compression, multimedia signal processing, information theory, and data security.

Dr. Mukherjee received the IEEE Student Paper award at the IEEE International Conference on Image Processing in Chicago, IL, in 1998.

Ricardo L. de Queiroz (S'86–M'94–SM'99) received the Engineering degree from the Universidade de Brasilia, Brasilia, Brazil, in 1987, the M.Sc. degree in electrical engineering from the Universidade Estadual de Campinas, Campinas, Brazil and the Ph.D. degree in electrical engineering from the University of Texas, Arlington, in 1994.

During 1990–1991, he was with the Digital Signal Processing Research Group at Universidade de Brasilia, as a Research Associate. He was with Xerox Corporation, Norwalk, CT, in 1994, where he was a Member of the Research Staff until 2002. During 2000–2001, he was also an Adjunct Faculty Member at the Rochester Institute of Technology, Rochester, NY. Currently, he is with the Departamento de Engenharia Eletrica, Universidade de Brasilia. He is the author of more than 100 articles in Journals and Conferences and contributed chapters to books as well. He also holds over 40 issued patents. His research interests include image and video compression, multirate signal processing, and color imaging.

Dr. de Queiroz is an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, and was an Editor for the IEEE SIGNAL PROCESSING LETTERS. He has been actively involved with the Rochester chapter of the IEEE Signal Processing Society, where he served as Chair and organized the Western New York Image Processing Workshop since its inception until 2001. He is the General Chair of IEEE International Symposium on Circuits and Systems 2011 and of multimedia signal processing 2009. He was also part of the organizing committee of International Conference on Image Processing 2002. He is a Member of the Brazilian Telecommunications Society and of the Brasilian Society of Television Engineers.