



biblio.ugent.be

The UGent Institutional Repository is the electronic archiving and dissemination platform for all UGent research publications. Ghent University has implemented a mandate stipulating that all academic publications of UGent researchers should be deposited and archived in this repository. Except for items where current copyright restrictions apply, these papers are available in Open Access.

This item is the archived peer-reviewed author-version of:

Title: Decoder-driven mode decision in a block-based distributed video codec

Authors: Stefaan Mys, Jürgen Slowack, Jozef Škorupa, Nikos Deligiannis, Peter Lambert, Adrian Munteanu, and Rik Van de Walle

In: Multimedia Tools and Applications, Volume 58, Number 1, pp. 239-266, May 2012.

To refer to or to cite this work, please use the citation to the published version:

Stefaan Mys, Jürgen Slowack, Jozef Škorupa, Nikos Deligiannis, Peter Lambert, Adrian Munteanu, and Rik Van de Walle (2010). Decoder-driven mode decision in a block-based distributed video codec. *Multimedia Tools and Applications*, Volume 58 (number 1), pages 239-266, May 2012. doi:10.1007/s11042-010-0718-5

Decoder-driven Mode Decision in a Block-Based Distributed Video Codec

Stefaan Mys · Jürgen Slowack ·
Jozef Škorupa · Nikos Deligiannis ·
Peter Lambert · Adrian Munteanu ·
Rik Van de Walle

Received: date / Accepted: date

Abstract Distributed Video Coding (DVC) is a video coding paradigm in which the computational complexity is shifted from the encoder to the decoder. DVC is based on information theoretic results suggesting that, under ideal conditions, the same rate-distortion performance can be achieved as for traditional video codecs. In practice however, there is still a significant performance gap between the two coding architectures. One of the main reasons for this gap is the lack of multiple coding modes in current DVC solutions.

In this paper, we propose a block-based distributed video codec that supports three coding modes: Wyner-Ziv, skip, and intra. The mode decision process is entirely decoder-driven. Skip blocks are selected based on the estimated accuracy of the side information. The choice between intra and Wyner-Ziv coding modes is made on a rate-distortion basis, by selecting the coding mode with the lowest rate while assuring equal distortion for both modes.

Experimental results illustrate that the proposed block-based architecture has some advantages over classical bitplane-based approaches. Introducing skip and intra coded blocks yields average bitrate gains of up to 33.7% over our basic configuration supporting Wyner-Ziv mode only, and up to 29.7% over the reference bitplane-based DISCOVER codec.

Keywords Skip Mode · Intra Mode · Mode decision · Distributed Video Coding · Wyner-Ziv coding

1 Introduction

Since the introduction of digital video in the late 1970s there has been a need for video compression, due to the limited capacity of storage devices and networks. Hence, re-

Stefaan Mys · Jürgen Slowack · Jozef Škorupa · Peter Lambert · Rik Van de Walle
Ghent University – IBBT, Department of Electornics and Information Systems (ELIS) – Multimedia Lab, Gaston Crommenlaan 8 bus 201, B-9050 Ghent, Belgium
E-mail: stefaan.mys@ugent.be

Nikos Deligiannis · Adrian Munteanu
Vrije Universiteit Brussel – IBBT, Electronics and Informatics Department (ETRO), Pleinlaan 2, B-1050 Brussels, Belgium

search towards video coding (or compression) has been a hot topic since. This research has led to several international coding standards, of which MPEG-x and H.26x are the best known and most successful. Typically, compression is achieved by exploiting the statistical redundancies present in the video sequence at the encoder. As a result, typical codecs consist of a computationally complex encoder and a fairly simple decoder. Such a setup suits applications in which the video sequence is encoded only once but decoded many times (as in broadcasting scenarios), for example.

However, other applications (e.g., wireless video surveillance or video conferencing with mobile devices) could benefit from the opposite situation, i.e., a low complexity encoder coupled with a more complex decoder [17]. Therefore, in the past decade, a coding paradigm called distributed video coding (DVC) [12, 18] has gained the attention of the scientific community.

In DVC, frames are coded independently from each other but decoded jointly. As the temporal redundancies are exploited by the decoder exclusively, in DVC the computational burden is shifted from the encoder to the decoder.

DVC is based on the information-theoretic results of Slepian and Wolf [21], and Wyner and Ziv [30]¹. Although these results suggest that – in ideal conditions – a DVC system could achieve the same rate-distortion performance as a traditional video coding system, in practice, the performance of all known DVC codecs is significantly inferior to conventional codecs such as H.264/AVC.

This performance gap is explained by several reasons. Firstly, performing motion estimation at the decoder instead of at the encoder inevitably results in a less accurate prediction signal (called side information in DVC), since at the decoder the original frame cannot be used to find the optimal motion vectors. Secondly, the results from Slepian and Wolf, and Wyner and Ziv assume that the correlation between the original frame X and the side information Y generated at the decoder is known in the form of a conditional probability mass function $p(X|Y)$. Since Y is not available at the encoder (and X is not available at the decoder) this conditional distribution needs to be estimated. Inaccuracies in the estimated correlation model result in suboptimal performance of the Wyner-Ziv coder.

Thirdly, traditional state-of-the-art video codecs employ a rich set of intra- and inter-prediction modes as well as advanced rate-distortion-driven mode selection mechanisms. This highly advanced mode decision process allows adapting to possibly varying characteristics in a video sequence. In contrast, current DVC systems do not employ sophisticated prediction modes. They often rely on only one coding mode (i.e., WZ coding) applied for the entire frame.

In this paper we tackle the third problem. We propose a block-based DVC codec able to encode WZ frames using one out of three coding modes, i.e., the intra, skip, or Wyner-Ziv mode. Mode decision is performed entirely at the decoder on a rate-distortion basis. In the following section of this paper we provide an overview of existing related work (Section 2). Next, the proposed codec is discussed from a high level point of view (Section 3). This is followed by a detailed description of the skip mode (Section 4.1) and the intra mode (Section 4.2), including a description of how the decoder chooses between them. Finally, experimental results are provided in Section 5 followed by conclusions in Section 6.

¹ Therefore, DVC is often referred to as Wyner-Ziv (WZ) coding, and the independently encoded/jointly decoded frames in a DVC codec are called WZ frames.

2 Related Work

The idea of using skip and/or intra coding to increase the performance of DVC is not new. Several related papers have appeared in the literature. This section provides an overview of the ones most relevant to our work.

Merely all recent DVC codecs are based on the architecture initially proposed by Aaron et al. [1]. In this codec, a video sequence is partitioned into key frames and Wyner-Ziv (WZ) frames. The key frames are intra coded and decoded independently from other frames. Wyner-Ziv frames are transformed by a discrete cosine transform (DCT), and quantized at the encoder. The quantized DCT coefficients are grouped into coefficient bands (e.g. the first coefficient band contains all DC coefficients) and bitplanes are extracted (e.g. the first bitplane contains all most significant bits of all DC coefficients). These bitplanes are fed as codewords to a channel coder (e.g., turbo or LDPC). The parity bits generated by the channel coder are stored in a buffer, and sent in portions to the decoder upon request.

At the decoder, side information is generated for each Wyner-Ziv frame based on two previously decoded (key or WZ) frames, i.e., one past and one future reference frame. Next, the side information is transformed through a DCT, and errors in the side information bitplanes are corrected by using the parity bits received from the encoder. The corrected bitplanes are regrouped, and the most likely coefficient within the corrected quantization bin is selected. Finally, the decoded frame is obtained after performing the inverse DCT.

Apart from the codec proposed by Aaron et al., a second pioneering architecture in DVC is the PRISM codec, developed by Puri et al. [19,20]. In PRISM, each frame is partitioned into blocks, and each block is classified into one of several classes (i.e., skip, WZ, or intra). Blocks classified as skip are not coded. In that case, the collocated block in the previous frame serves as the decoded result. Blocks classified as Wyner-Ziv are first transformed. Next, only the low frequency coefficients are Wyner-Ziv coded (through syndrome coding), while the high frequency coefficients are intra coded. The PRISM architecture proved far less popular than the codec by Aaron et al. and had few followers in literature. As a result, in terms of rate-distortion performance, the PRISM architecture is outdated and outperformed by current designs stemming from the codec proposed by Aaron et al.

These two pioneering architectures have introduced roughly two classes of DVC codecs, namely *bitplane-based codecs* and *block-based codecs*. In *bitplane-based codecs*, bits are grouped according to their frequency and index. *Blocks-based codecs* such as PRISM on the other hand, partition frames into blocks and group the bits in each block. In this case, the codewords serving as input to the channel coder consist of all bits from all transformation coefficients of a certain block of pixels.

Chien and Karam [7] developed a rate-distortion model which is used at the decoder to decide which bitplanes to decode and which to skip. Their codec yields competitive performance for sequences with low to medium motion. However, this system does not seem to perform well for sequences exhibiting moderate to high motion. This is presumably caused by the fact that, by skipping entire bitplanes, the codec is not able to adapt to the spatially varying quality of the side information (generated at the decoder).

Therefore, Belkoura and Sikora [5], and Feng et al. [11] propose to skip blocks of pixels (i.e., in the spatial direction) instead of frequency-related bitplanes. The decision to skip a block is made at the encoder based on the mean squared error between a block

and its collocated block in the previous frame [5], or at the decoder based on the sum of squared errors between the reference blocks used to generate the side information [11]. In both contributions a bitplane-based codec is used. The bits that should be skipped are either discarded at the encoder [5], or replaced by zeros [11]. Both approaches have a negative impact on the efficiency of the channel coder.

Our previous work [16] presents a way to avoid inefficient channel coding while still employing skipped blocks. In this approach, blocks marked as skipped by the encoder are not removed from the encoding phase and parity bits are still calculated over the entire frame. At the decoder, knowledge about the skip mode of each block is exploited during channel decoding instead, as well as in the puncturing of the parity bits. This results in significant rate gains.

Chien, Karam and Arousleman [8] propose to scan the bitplanes blockwise instead of row-by-row, allowing the decoder to adaptively change the number of requested parity bits on a block-per-block basis. By allowing the number of parity bits to be zero, skipped blocks are indirectly supported.

Do et al. [9] perform mode decision at the decoder based on an evaluation of the linearity of the motion vectors. Blocks with linear motion vectors are skipped. In the case of highly non-linear motion vectors, additional hash information is requested from the encoder to help improving the quality of the side information.

Esmaili and Cosman [10] add skip and Wyner-Ziv coded blocks to the key frames. At the encoder, the mean squared error between each block in the key frame and the collocated block in the previously decoded key frame is computed. Blocks with a mean squared error below a certain threshold are skipped, while blocks with a mean squared error above a second threshold are intra coded. The remaining blocks are Wyner-Ziv coded. In order to keep the length of the codewords used as input to the channel coder fixed, the bits corresponding to skip or intra blocks are set to zero.

Trapanese et al. [26] introduce intra coded blocks in the Wyner-Ziv frames of a bitplane-based codec. Based on a threshold on the sum of the absolute differences between two blocks, some blocks are marked as intra. These blocks are intra coded and all bits corresponding to intra coded blocks are skipped when extracting the codewords for the Wyner-Ziv coder. The intra mode decision can be made either at the encoder or at the decoder. In an extension of their work [25] they propose to use an additional spatial smoothness metric to decide upon the coding mode. Similar criteria are used by Tsai et al. [27] in a block-based DVC codec in which key frames are replaced by intra coded blocks in the Wyner-Ziv frames.

Benierbah and Khamadja [6] also propose a block-based codec without key frames. In this case the frames are divided into Wyner-Ziv blocks and intra blocks following a checkerboard pattern. The decoded intra blocks are used at the decoder to help generating the side information for the remaining WZ blocks, as well as to estimate their reliability. Due to the fact that a fixed pattern is used for the modes, no mode information needs to be transmitted. On the other hand, spatial properties in the video are not accounted for, making the mode decision suboptimal.

Ascenso et al. [4] propose a combined Intra/WZ coding mode. For certain blocks, a low-quality intra coded version is sent from the encoder to the decoder, to help improving the quality of the side information. In order to choose between Intra/WZ and normal WZ mode, the encoder estimates and minimizes the required rate.

Table 1 summarizes some important properties of the discussed papers. It points out three important drawbacks shared by several of the papers. Firstly, many of the proposed codecs are bitplane-based codecs. As a result, in order to skip or intra code

Table 1 Summary of related work.

| | encoder(E) or decoder(D) driven | plane(P) or block(B) based | rate- distortion based | skip mode | intra mode | comments |
|-------|--|-------------------------------------|------------------------------|--------------|---------------|----------------------|
| [7] | D | P | • | • | | skipped planes |
| [5] | E | P | | • | | |
| [11] | D | P | | • | | |
| [16] | E | P | | • | | residual codec |
| [8] | E | B | | • | | |
| [9] | E | P | | • | | extra hash mode |
| [10] | E | P | | • | • | in key frames |
| [26] | E/D | P | | • | • | |
| [25] | E/D | P | | • | • | |
| [27] | E | B | | • | • | |
| [6] | n/a | B | | | • | checkerboard pattern |
| [4] | E | P | • | | | intra/wz mode |
| prop. | D | B | • | • | • | |

at the block level, special precautions need to be taken to construct the codewords for the channel coder. Either the codewords will not have a constant length, or bits corresponding to skip or intra blocks will have to be replaced by zeros. Both approaches have a negative impact on the efficiency of the decoder.

Secondly, often the mode decision is performed at the encoder. This does not fit well the DVC paradigm since it adds complexity to the encoder. Also, the encoder-driven mode decision cannot take the quality of the side information into account when performing the mode decision. It can be noticed from the results in [25,26] that this does affect the rate-distortion performance negatively.

Thirdly, only two of the enumerated papers directly use rate-distortion metrics for performing mode decision. The other contributions apply thresholds on various metrics, indirectly related to the estimated required WZ rate. Provided that the rate and distortion estimations are accurate, using them directly to decide between the coding modes is more appropriate and straightforward, and it should therefore lead to a more accurate mode decision.

To avoid these three disadvantages this paper proposes a block-based codec implementing skip, Wyner-Ziv, and intra mode, in which the mode decision is performed at the decoder on a rate-distortion basis. The next section will describe the proposed codec in detail.

3 The proposed block-based DVC codec with decoder-driven mode decision

Figure 1 shows the proposed block-based DVC codec with decoder-driven mode decision, and an overview of the interaction between the encoder and the decoder is provided in Figure 2. The coding process starts with the encoding of two key frames, which are intra coded using H.264/AVC [29] and which can be decoded independently from other frames.

Once a past and a future reference frame are decoded, the decoder generates the side information for an intermediate WZ frame. The methodology proposed by Ascenso et al. [3] for generating side information is used. Next, the virtual noise between the side

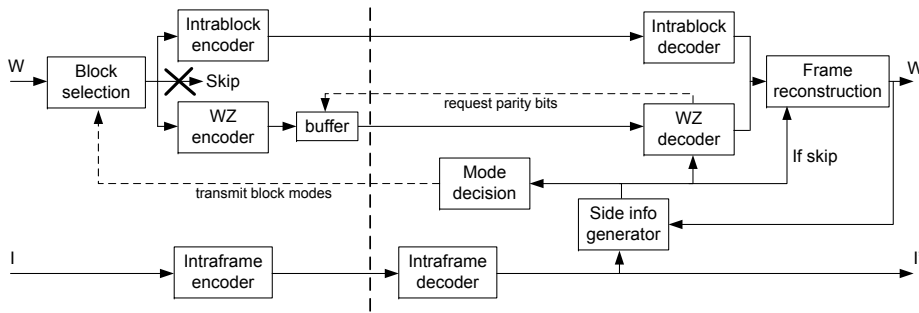


Fig. 1 The proposed block-based DVC codec with decoder-driven mode decision.

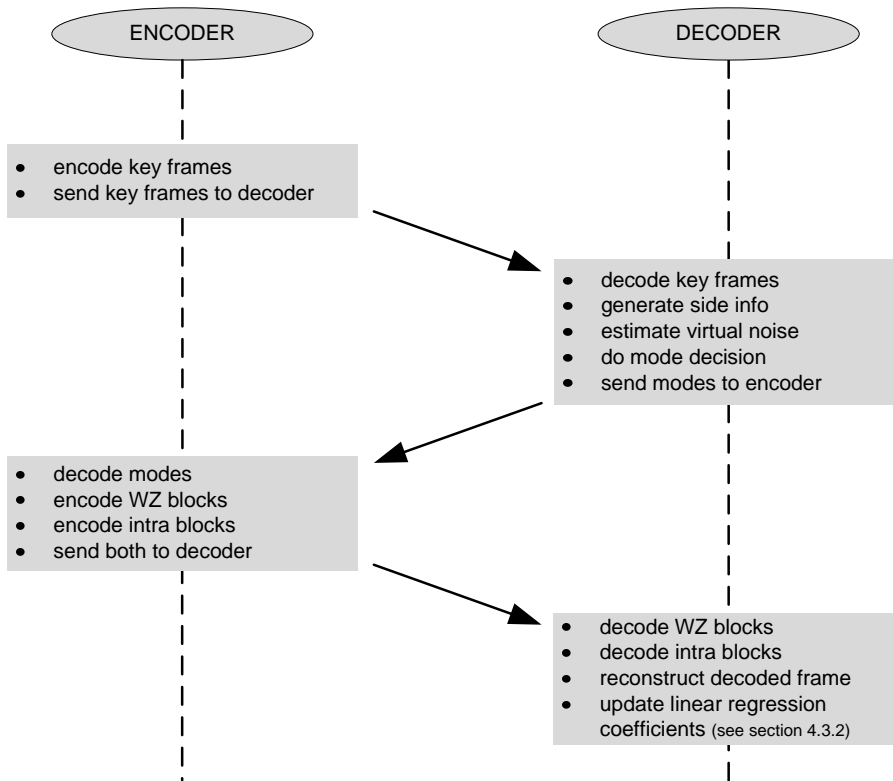


Fig. 2 Encoder/decoder interaction in the block-based DVC codec with decoder-driven mode decision.

information and the original frame is estimated, following our approach described in [22]. Then, the frame is divided into 16×16 macroblocks and for each block the coding mode is determined by the mode decision module at the decoder. The coding modes are encoded using adaptive arithmetic coding and transmitted to the encoder.

The encoder groups all intra blocks into one slice which is encoded by the H.264/AVC intra coder applying random macroblock ordering. As a result, intra prediction from

| | | | | | | | | | | | |
|-----------------|----|----|----|-----------------|----|----|----|-----------------|----|----|----|
| 15 | 11 | 08 | 06 | 27 | 13 | 08 | 04 | 60 | 12 | 04 | 01 |
| 11 | 08 | 06 | 04 | 13 | 08 | 04 | 02 | 12 | 04 | 01 | 00 |
| 08 | 06 | 04 | 02 | 08 | 04 | 02 | 01 | 04 | 01 | 00 | 00 |
| 06 | 04 | 02 | 01 | 04 | 02 | 01 | 00 | 01 | 00 | 00 | 00 |
| (a) Q1 (7 bits) | | | | (b) Q3 (5 bits) | | | | (c) Q5 (3 bits) | | | |

Fig. 3 Distribution of the parity bit rate over the transform coefficients for three different quantization levels. The numbers denote the percentage of the total rate spent at the corresponding transform coefficients.

neighboring blocks can be used provided that these neighboring blocks are also intra coded blocks. Skip blocks are discarded and take no further part in the encoding process. All other blocks are Wyner-Ziv blocks. One by one they are transformed and quantized in the same way as in our previous work [16]. The quantized transform coefficients of each block are grouped into a single codeword, with the bits in the codeword ordered according to the transform coefficient they belong to. Finally a turbo coder calculates parity bits for the codeword and stores them in a buffer.

At the decoder, the intra coded blocks are decoded by the H.264/AVC intra decoder. The WZ blocks are decoded by the WZ decoder, which corrects errors in the side information using parity bits requested from the encoder's buffer. The amount of parity bits to be requested is determined by gradually increasing the rate until the turbo decoder is able to correct all errors in the side information². For skipped blocks the side information is used as the decoded block. Finally, in the frame reconstruction module all decoded blocks are combined to form the decoded frame.

Puncturing is applied in order to determine at the encoder, for a given requested rate, which parity bits to transmit and which to discard. Typically, bits are punctured following a pseudo-random pattern. Although in theory each parity bit contains information about each input bit in the codeword, in practice, the decoding process associated with a certain bit is mainly influenced by the information from surrounding parity bits. Therefore, in our codec, the punctured bits are not uniformly distributed over the entire codeword. Instead, the lower the frequency band the coefficient belongs to (in zigzag scan order), the less parity bits are punctured. Per quantization level, a fixed, experimentally derived distribution of the parity rate over the coefficients is applied. Examples of these distributions for three quantization patterns are provided in Figure 3.

Note that, although the coding modes (intra, skip or Wyner-Ziv) are assigned to 16x16 macroblocks, the actual block sizes used in the codec can vary. In the case of intra, any of the available intra coding modes in H.264/AVC can be applied. This can be one of the four available 16x16 intra prediction modes, or a combination of four of the nine available 4x4 intra coding modes. This decision is taken in the rate-distortion optimized mode decision module of the H.264/AVC intra coder. For Wyner-Ziv macroblocks, in the side information process a motion vector is assigned to each 8x8 macroblock partition. The side information generation process, including the block sizes, is taken directly from [3] and is identical to the one used in the DISCOVER codec.

² Although online stopping criteria for the turbo decoder have been described in the literature [2, 14, 28, 24], in this case perfect error detection at the decoder is assumed.

Table 2 Average MSE for blocks with and without errors in the side information. Rows are ordered from fine to coarse quantization.

| | Foreman | | Table Tennis | | Mother and Daughter | | Chosen threshold |
|---|-----------|--------|--------------|--------|---------------------|--------|------------------|
| | no errors | errors | no errors | errors | no errors | errors | |
| 1 | 0.20 | 95.81 | 0.79 | 136.76 | 0.17 | 17.03 | 0.8 |
| 2 | 0.81 | 95.75 | 1.26 | 137.43 | 0.35 | 19.06 | 2 |
| 3 | 2.26 | 101.82 | 3.05 | 141.41 | 0.98 | 21.24 | 5 |
| 4 | 6.08 | 114.82 | 5.18 | 187.61 | 3.57 | 24.99 | 15 |
| 5 | 17.15 | 130.41 | 11.09 | 259.12 | 8.11 | 31.41 | 28 |

4 The coding modes and the mode decision

4.1 Skip Mode

The goal of the skip mode is to detect blocks for which the quality of the side information is good enough so that it can be used as a decoded block. This decision is based on the mean squared error (MSE) between the past and the future reference block used to create the side information. If the MSE is small, the side information is assumed to be reliable, and the block is skipped. If not, the block is either intra coded or Wyner-Ziv coded (see Section 4.2).

A quantization level dependent threshold is used to determine if a block should be skipped or not. From coarse to fine quantization, the following cutoff values (i.e., thresholds) are defined: 0.8, 2, 5, 15, and 28. If the MSE is below these values the block is skipped. The same values are used for all sequences.

These cutoff values have been chosen based on the results of offline experiments. For several sequences, side information was generated and all macroblocks in the side information were divided into two sets. One set contained the blocks for which, after transformation and quantization, the side information was identical to the original frame. The second set contained all macroblocks with side information that, after transformation and quantization, differed at least one bit with the original frame. Table 2 lists the average MSE for both sets, indicating that the differences between the sets are significant. Therefore, a general cutoff value over all sequences is expected to perform well. Based on these experimental results, the aforementioned cutoff values were chosen.

Table 3 shows the precision and recall³ corresponding to the chosen thresholds. To calculate the precision and recall, the ground truth was set so that only blocks containing no errors are skipped. However, as the results in Section 5 will demonstrate, sometimes it can be beneficial to skip blocks even if the side information contains a few errors. This is especially true for low rates and for sequences containing low motion. Therefore, the low precision values are not necessarily harmful, and the thresholds were chosen primarily to keep the recall high (i.e. to assure that almost all blocks that should be skipped, will be skipped). Also note that the low number of positives for high rates makes the low precision values less relevant.

³ precision = $\frac{\text{true positives}}{\text{true positives} + \text{false positives}}$; recall = $\frac{\text{true positives}}{\text{true positives} + \text{false negatives}}$

Table 3 Precision and recall results for the skip mode decision. Rows are ordered from fine to coarse quantization. The first column indicates the number of positives, i.e., the percentage of error-free blocks in the side information.

| | Nr. positives (%) | Precision (%) | Recall (%) |
|---|-------------------|---------------|------------|
| 1 | 1.7 | 14.8 | 97.6 |
| 2 | 5.5 | 28.5 | 96.4 |
| 3 | 11.9 | 35.6 | 92.5 |
| 4 | 25.2 | 44.2 | 93.1 |
| 5 | 37.0 | 55.5 | 90.8 |

4.2 Intra Mode versus Wyner-Ziv mode

Macroblocks that are not skipped are either intra or Wyner-Ziv coded. This decision is based on an estimation of the rate and the distortion in both cases. First (Section 4.2.1), it is explained how it can be assured that the average distortion for intra coded blocks is the same as the average distortion for Wyner-Ziv coded blocks. Considering this being the case, the coding mode requiring the lowest rate is then chosen as the rate-distortion optimal coding mode. In order to know which mode requires the lowest rate, the required rate for both coding modes is estimated. Sections 4.2.2 and 4.2.3 explain how this is done.

4.2.1 Distortion control

For the distortion, the coding parameters are chosen such that the quality of the decoded intra blocks matches the quality of the decoded Wyner-Ziv blocks. This is done in a two-step process, as follows.

Firstly, the intra quantization parameter (intra QP) that will be used to code the key frames is chosen such that the quality of the decoded key frames matches the quality of the decoded Wyner-Ziv frames (applying the Wyner-Ziv mode only). This is a common assumption in DVC that could be achieved online [13,23]. However, in this paper, the intra QPs of the key frames are determined (per sequence and per quantization level) in an offline setup prior to the actual coding process.

In a second step, we determine the intra QP for the macroblocks classified as intra by the mode decision algorithm proposed in this paper. Note that simply using the intra QP of the key frames to encode these blocks would not lead to the desired result. Since, unlike in the key frames, the intra coded blocks in a Wyner-Ziv frame are in general not adjacent. As a result, intra prediction will be less efficient, and so using the same QP would most likely lead to a result of lower quality. Therefore, before encoding the intra blocks in each Wyner-Ziv frame, the encoder determines the quality of the previously encoded key frame. This is possible since both the original and the decoded key frame are available at the encoder. Subsequently, it encodes the intra blocks in the Wyner-Ziv frame using a QP chosen in such a way that the quality of the decoded intra blocks is similar to the quality of the decoded key frame.

Since the quality of the decoded intra blocks matches the quality of the decoded key frame, and the quality of the decoded key frame matches the quality of the decoded Wyner-Ziv blocks, it can be concluded that the quality of the decoded Wyner-Ziv blocks should be similar to the quality of the decoded intra blocks. This statement will be evaluated in Section 5.

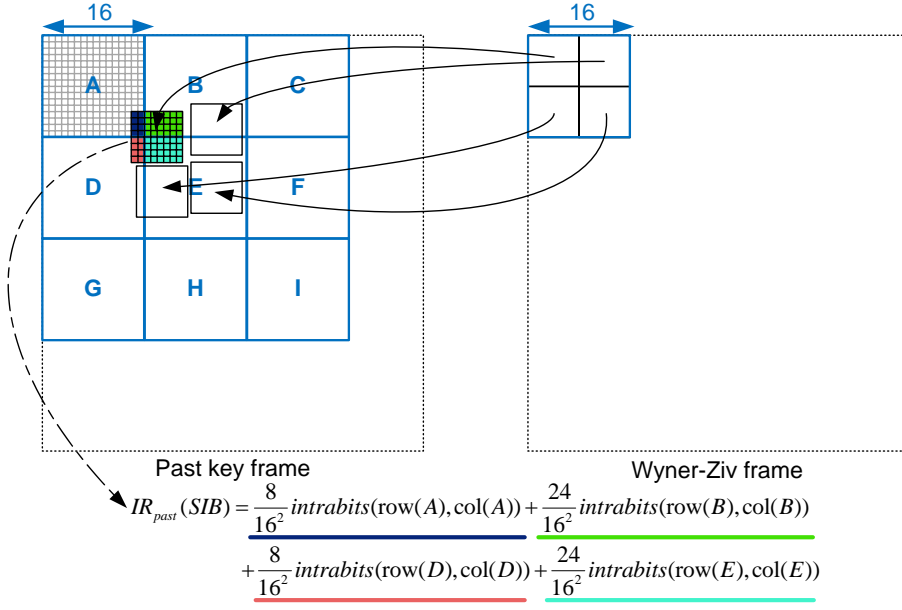


Fig. 4 Illustration of equations (2) and (3), used to estimated the required intra rate for a macroblock.

4.2.2 Intra rate estimation

The required intra rate is estimated based on the intra rate used to code the past and future key frames. More precisely, a weighted average between an estimation based on the past key frame (R_{intra}^{past}) and an estimation based on the future key frame (R_{intra}^{future}) is used. The weights are chosen based on the distance between the current frame and the respective key frame. Let G be the distance (in frames) between the two key frames, and d_{past} (resp. d_{future}) the distance between the current frame and the nearest past (resp. nearest future) key frame. Then the estimated intra rate is written as:

$$R_{intra} = \frac{(G - d_{past})}{G} R_{intra}^{past} + \frac{(G - d_{future})}{G} R_{intra}^{future}. \quad (1)$$

The calculation of R_{intra}^{past} will be described in detail in the remainder of this section. An example is also provided in Figure 4 as additional support to the reader. The calculation of R_{intra}^{future} is completely analogously.

Firstly, R_{intra}^{past} is defined as the sum of the estimated intra rate IR_{past} for each of the side information blocks SIB^4 within the macroblock:

$$R_{intra}^{past} = \sum_{SIB} IR_{past}(SIB). \quad (2)$$

Since macroblocks are 16×16 pixels, and side information blocks are 8×8 pixels, this summation goes over 4 side information blocks SIB .

⁴ By side information block, we refer to the blocks to which a bidirectional motion vector was assigned during the side information generation. In the proposed codec, side information blocks are 8×8 pixels.

$IR_{past}(SIB)$ is calculated as follows. Let $SIB_y^{topleft}$ and $SIB_x^{topleft}$ be the pixel row and column of the topleft pixel in the side information block SIB , and let y and x be the pixel row and column within SIB . Furthermore, let SIB_y^{mv} and SIB_x^{mv} denote the vertical and horizontal components of the motion vector obtained by extrapolating the motion vector from SIB towards the nearest past key frame. Then, $IR_{past}(SIB)$ is calculated as follows:

$$IR_{past}(SIB) = \sum_{y=0}^7 \sum_{x=0}^7 \frac{1}{16^2} intrabits \left(\left\lfloor \frac{SIB_y^{topleft} + y + SIB_y^{mv}}{16} \right\rfloor, \left\lfloor \frac{SIB_x^{topleft} + x + SIB_x^{mv}}{16} \right\rfloor \right), \quad (3)$$

where $intrabits(macrobloc_row, macrobloc_col)$ returns the intra rate spent on the macroblock (of size 16×16) at row $macrobloc_row$ and column $macrobloc_col$ in the nearest past key frame. Due to the fact that we iterate over all pixels, the factor $1/16^2$ is used to normalize the result of $intrabits$ to a pixel average. The calculation of $IR_{past}(SIB)$ is included in the example depicted in Figure 4.

Finally, the difference between the QP used to code the key frames and the QP used to code the intra blocks is taken into account. For each step the intra block QP is lower than the key frame QP, the estimated rate is increased by 12%, and vice versa [15].

4.2.3 Wyner-Ziv rate estimation

The Wyner-Ziv rate is estimated using a linear regression function based on two criteria, i.e.:

$$R_{WZ} = c_0 \cdot P_0(MB) + c_1 \cdot P_1(MB). \quad (4)$$

The first parameter, P_0 , is related to the quality of the side information. This is taken into account by means of the MSE between the past and future reference blocks. Since we want to know the local quality of the side information, instead of directly using the MSE we use the deviation between the MSE for the block under consideration and the average MSE for all blocks in the side information. Hence, P_0 is given by

$$P_0(MB) = \left| MSE(MB) - \frac{1}{B} \sum_{i=0}^{B-1} MSE(MB_i) \right|, \quad (5)$$

where B denotes the total number of macroblocks in this frame, MB_i denotes the i^{th} macroblock, and $MSE(b)$ denotes the mean squared error between the past and future reference blocks.

The second parameter P_1 is similar to the intra rate estimation. It gives the number of Wyner-Ziv bits spent for the corresponding block in the previously decoded frame. First, we define P_1 as the sum of the estimated Wyner-Ziv rate WR of the four side information blocks SIB contained in the macroblock:

$$P_1(MB) = \sum_{SIB} WR(SIB). \quad (6)$$

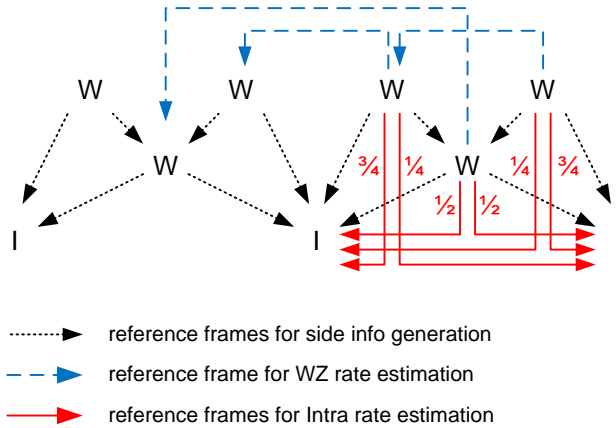


Fig. 5 Reference frames for side info generation and rate estimation in a GOP of 4 frames.

In turn, the estimated Wyner-Ziv rate WR for each of the side information blocks is calculated as follows:

$$WR(SIB) = \sum_{y=0}^7 \sum_{x=0}^7 \frac{1}{16^2} wzbts \left(\left[\frac{SIB_y^{topleft+y} + SIB_y^{mv}}{16} \right], \left[\frac{SIB_{topleft_x+x} + SIB_x^{mv}}{16} \right] \right). \quad (7)$$

In this formula, $wzbts(\text{macroblock_row}, \text{macroblock_col})$ returns, per 16×16 macroblock, the number of Wyner-Ziv bits spent on that block in the previously decoded frame. For blocks that were skipped in the previously decoded frame $wzbts$ equals 0. For intra blocks, an encoder loop is added to the decoder. After decoding the intra block, the decoded block is Wyner-Ziv coded and decoded, and the required Wyner-Ziv bits are stored in $wzbts$.

An important remark needs to be made about the term ‘*previously decoded frame*’. Since the quality of the side information and thus also the required Wyner-Ziv rate depends on the distance between the frame and the reference frames, by ‘previously decoded frame’ we refer to the previously decoded frame in the same hierarchical layer. For example, for a GOP of 4 frames (see Fig. 5), the previously decoded frame for the middle WZ frame in a GOP refers to the middle WZ frame in the previous GOP; the previously decoded frame for the first WZ frame refers to the last WZ frame in the previous GOP; and the previously decoded frame for the last WZ frame in a GOP refers to the first WZ frame of that same GOP. As a result, to obtain the motion vector mv used in (7), the backward motion vector from the block to which the pixel on position (x, y) belongs is doubled.

The coefficients c_0 and c_1 in equation 4.2.3 are determined through linear regression. Each time a frame is decoded, the coefficients are updated using least absolute deviation, i.e., minimizing the sum of the absolute deviations of the errors:

$$\{c_0, c_1\} = \underset{\{c_0, c_1\}}{\operatorname{argmin}} \sum_{b=0}^{M-1} |y(b) - (c_0 \cdot P_0(b) + c_1 \cdot P_1(b))| \quad (8)$$

where $y(b)$ is the actual number of Wyner-Ziv bits spent, P_0 and P_1 are given above, and M is the number of macroblocks taken into account for the minimization.

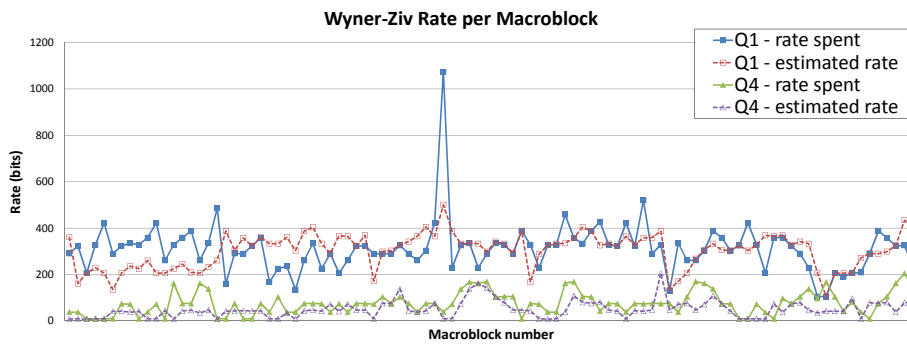


Fig. 6 Estimated and actual Wyner-Ziv rate per macroblock. Table Tennis sequence, CIF, 30 fps.

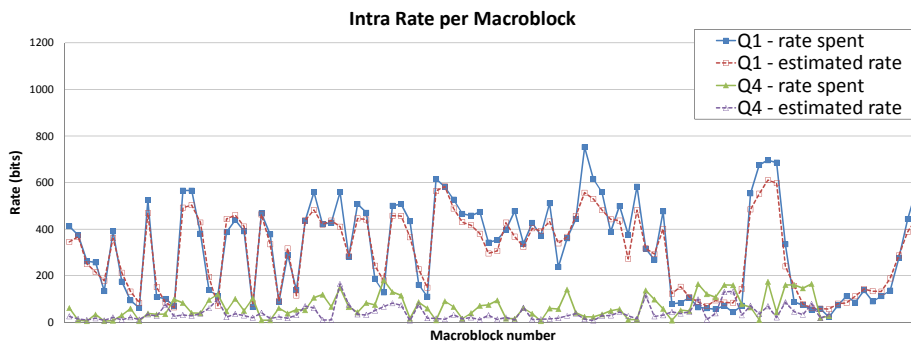


Fig. 7 Estimated and actual intra rate per macroblock. Table Tennis sequence, CIF, 30 fps.

Two different sets of coefficients c_0 and c_1 are maintained, each corresponding to a hierarchical layer⁵, and in each update all non-skip macroblocks from the last two⁶ decoded frames in the current hierarchical layer are taken into account. Thus, the coefficients c_0 and c_1 used to estimate the Wyner-Ziv rate for blocks in a certain frame at position F are obtained by minimizing the sum of the absolute deviations of the estimation errors for all non-skip blocks in the two previously decoded frames at positions $F - 1$ and $F - 2$ in the same hierarchical layer.

4.2.4 Rate estimation performance

Figures 6 and 7 show the performance of the Wyner-Ziv and intra rate estimation for some selected macroblocks. The fluctuations in the graph illustrate the advantage of working block-based instead of bitplane-based: the rate (and coding mode) varies from block to block to adapt to the spatial properties of the video. As expected, the error is larger for the Wyner-Ziv rate estimator than for the intra rate estimator. Still, for the majority of the blocks, the estimated rate follows quite accurately the actual rate, enabling accurate mode decision.

⁵ This is valid for a GOP size 4. More sets will be required for longer GOPs.

⁶ Experiments showed that considering two frames yields good results. Adding more frames does not improve the performance.

Table 4 Average bitrate gain (Bjontegaard metric (%)) of the proposed codec. The four lowest quantization levels (Q5,Q4,Q3 and Q2) are considered.

| | Relative to DISCOVER | | Relative to WZ mode only WZ + skip + intra mode |
|---------------------|----------------------|---------------------------|---|
| | WZ mode only | WZ + skip + intra mode | |
| Foreman | 8.3 | 21.4 | 12.0 |
| Table Tennis | 17.3 | 29.7 | 12.1 |
| Mother and Daughter | -1.0 | 14.4 | 33.7 |

Table 5 Number of blocks (%) coded in each coding mode when using online mode decision. Rows are ordered from fine to coarse quantization.

| | Foreman | | | Table Tennis | | | Mother and Daughter | | |
|---|---------|------|-------|--------------|------|-------|---------------------|------|-------|
| | WZ | skip | intra | WZ | skip | intra | WZ | skip | intra |
| 1 | 53.6 | 3.6 | 42.8 | 63.6 | 1.5 | 34.9 | 65.4 | 28.4 | 6.2 |
| 2 | 68.4 | 6.9 | 24.6 | 71.7 | 7.3 | 21.0 | 56.7 | 41.9 | 1.4 |
| 3 | 58.4 | 19.9 | 21.7 | 68.6 | 19.9 | 11.5 | 45.2 | 53.4 | 1.4 |
| 4 | 56.2 | 35.6 | 8.2 | 27.5 | 56.9 | 15.5 | 23.7 | 75.8 | 0.5 |
| 5 | 49.6 | 45.6 | 4.8 | 29.1 | 62.7 | 8.2 | 15.7 | 83.4 | 0.9 |

Table 6 Quality of the decoded Wyner-Ziv blocks and the decoded Intra blocks (yPSNR (dB)).

| | Foreman | | Table Tennis | | Mother and Daughter | |
|---|--------------|------------------|--------------|------------------|---------------------|------------------|
| | Intra blocks | Wyner-Ziv blocks | Intra blocks | Wyner-Ziv blocks | Intra blocks | Wyner-Ziv blocks |
| 1 | 42.11 | 41.92 | 41.98 | 41.83 | 44.38 | 43.89 |
| 2 | 38.58 | 38.00 | 37.57 | 37.07 | 42.36 | 40.44 |
| 3 | 34.76 | 34.10 | 33.93 | 32.72 | 38.79 | 36.59 |
| 4 | 31.36 | 30.30 | 27.77 | 28.28 | 36.44 | 32.92 |
| 5 | 28.23 | 26.86 | 25.43 | 24.73 | 30.86 | 29.06 |

Table 7 Number of blocks (%) coded in each coding mode when using offline mode decision. Rows are ordered from fine to coarse quantization.

| | Foreman | | | Table Tennis | | | Mother and Daughter | | |
|---|---------|------|-------|--------------|------|-------|---------------------|------|-------|
| | WZ | skip | intra | WZ | skip | intra | WZ | skip | intra |
| 1 | 44.8 | 0.5 | 54.7 | 61.8 | 0.03 | 38.2 | 68.9 | 4.7 | 26.4 |
| 2 | 63.8 | 1.7 | 34.5 | 72.4 | 0.6 | 27.0 | 71.9 | 13.9 | 14.2 |
| 3 | 62.2 | 6.5 | 31.3 | 69.7 | 4.6 | 25.6 | 66.4 | 23.8 | 9.9 |
| 4 | 62.2 | 17.4 | 20.4 | 16.0 | 19.6 | 64.4 | 55.5 | 37.2 | 7.4 |
| 5 | 56.4 | 26.5 | 17.1 | 17.4 | 35.6 | 47.0 | 36.9 | 44.2 | 18.8 |

5 Results and Discussion

5.1 Experimental setup

Four different versions of the codec proposed in this paper are evaluated. First, intra and skip mode are left out, and the WZ-only block-based codec is compared to the bitplane-based DISCOVER codec [2]. Next, skip (resp. intra) blocks are added and the influence on the coding performance is discussed. Finally, all modes are enabled and the system is compared to H.264/AVC inter and intra coding, DISCOVER, and the so-called Blast DVC codec.

Table 8 Feedback channel rate using online mode decision with all coding modes available. Rows are ordered from fine to coarse quantization.

| (a) Foreman | | | | |
|-------------|-------------------------------|---------------------------|------------------------------------|-------------------------------|
| | Requesting parity bits (kbps) | Transmitting modes (kbps) | Total feedback channel rate (kbps) | Percentage of actual rate (%) |
| 1 | 7 | 36 | 42 | 1 |
| 2 | 8 | 35 | 44 | 2 |
| 3 | 5 | 35 | 39 | 4 |
| 4 | 4 | 28 | 31 | 7 |
| 5 | 2 | 25 | 27 | 11 |

| (b) Table Tennis | | | | |
|------------------|-------------------------------|---------------------------|------------------------------------|-------------------------------|
| | Requesting parity bits (kbps) | Transmitting modes (kbps) | Total feedback channel rate (kbps) | Percentage of actual rate (%) |
| 1 | 10 | 38 | 48 | 1 |
| 2 | 10 | 37 | 47 | 2 |
| 3 | 7 | 34 | 41 | 3 |
| 4 | 1 | 30 | 31 | 7 |
| 5 | 1 | 27 | 28 | 12 |

| (c) Mother and Daughter | | | | |
|-------------------------|-------------------------------|---------------------------|------------------------------------|-------------------------------|
| | Requesting parity bits (kbps) | Transmitting modes (kbps) | Total feedback channel rate (kbps) | Percentage of actual rate (%) |
| 1 | 6 | 28 | 34 | 2 |
| 2 | 4 | 21 | 25 | 3 |
| 3 | 2 | 19 | 21 | 4 |
| 4 | 1 | 12 | 13 | 6 |
| 5 | 1 | 10 | 11 | 11 |

For the results that will be discussed first, several test sequences at CIF resolution (30 fps) have been coded. A GOP of size 4 was used, corresponding to a prediction structure as depicted in Figure 5. All sequences were coded using 5 different quantization levels, Q_5 to Q_1 , corresponding to 3 to 7 bits per transform coefficient, respectively. The rate-distortion curves are shown in Figure 8. Only luma rate and luma PSNR are considered to allow for a fair comparison with DISCOVER.

5.2 Performance of the basic block-based codec

Our basic configuration with WZ blocks only outperforms DISCOVER for medium to high bitrates. At low bitrates, DISCOVER performs better. DISCOVER also scores better for Mother and Daughter, containing little movement.

The major difference between this configuration (without skip and intra blocks) and DISCOVER is the way codewords are extracted. In the proposed block-based codec, it is possible to spend less rate on spatial blocks with accurate side information and more rate on blocks containing many side information errors. Thus, the proposed codec takes advantage of the fact that the accuracy of the side information can be spatially non-stationary. DISCOVER on the other hand is a plane-based codec, having the advantage that less rate can be spent on bitplanes that have less errors (e.g. bitplanes of high

frequency coefficient bands). This advantage is partially exploited in the proposed block-based codec by adjusting the puncturing procedure (as described in Section 3). Presumably, more efficient techniques than the basic approach applied in this paper could be developed, boosting the general rate-distortion performance of the block-based codec. However, this requires further investigation which falls out of the scope of this paper. The proposed codec also uses a more advanced virtual noise estimator than DISCOVER, taking the intra quantization noise into account [22].

5.3 Performance of the skip and intra modes

Limited performance gain is achieved when adding the skip mode to the codec. This is because in the block-based codec each block will require only the necessary rate to correct the errors in the side information. Therefore, blocks containing no errors in the side information (which should be skipped if skip mode is enabled) will spend little or no rate in the Wyner-Ziv coding mode as well. However, for the Mother and Daughter sequence and for the low bitrates in the Table Tennis and Foreman sequences, the skip mode does bring a performance gain. The reason for this is that blocks are skipped which are not entirely error-free. In that case the slight decrease in quality caused by not correcting these errors may be outweighed by the more significant rate gain that can be achieved. This proves that the low results for the precision of the skip mode (Section 4.1 Table 3) do not necessarily have a negative impact on the actual rate-distortion performance.

For the Table Tennis and Foreman sequences, especially at medium to high rates, a significant performance gain is achieved by adding the intra mode. For Mother and Daughter, no gain is achieved, and for middle to high rates even small performance losses can be observed. Since the Mother and Daughter sequence contains low motion, the skip and Wyner-Ziv coding modes perform already very well. Therefore, only very few blocks will be selected as intra blocks, and even with perfect rate estimators only a small gain would be achieved. Suboptimal rate estimation causes the small performance loss in this case.

Not surprisingly, the best results are achieved when combining all three coding modes. For the low rates or low motion sequences, the achievable gains by the skip mode are exploited. For higher rates and for sequences containing irregular motion, many intra blocks will be chosen to achieve good rate-distortion performances. Depending on the sequence, average bitrate gains up to 33.7% can be achieved compared to the basic block-based codec with Wyner-Ziv mode only, or up to 29.7% compared to DISCOVER (see Table 4).

Table 5 shows the number of blocks coded using each coding mode when using the proposed online mode decision. As expected, the number of skip blocks increases by decreasing rate, while at the same time the number of intra blocks decreases. Also, sequences exhibiting low motion (Mother and Daughter), employ more skip and less intra blocks compared to more motion-heavy sequences (Table Tennis and Foreman). Figure 5 shows an example frame from the Table Tennis sequence which illustrates how the three modes are assigned to the different blocks.

Table 6 reports the quality of the decoded Wyner-Ziv blocks and the quality of the decoded intra blocks. Ideally, both qualities should be the same, as discussed in Section 4.2. The deviations that can be observed, especially at low bitrates, are caused by the first step of the process described in Section 4.2. In that step, the offline procedure

matches the average quality of the key frames to the average quality of the Wyner-Ziv frames. However, since the same QP is used to code all key frames, some frame-to-frame variations in quality are inevitable.

5.4 Comparison with offline mode decision

To evaluate the cost of mode decision inaccuracies, the intra and Wyner-Ziv coding modes are compared with a codec performing perfect (but offline) mode decision. For the skip mode, perfect mode decision is achieved by comparing the side information for each block with the original frame, and skipping the block only if no errors occur. For intra versus Wyner-Ziv mode decision, each block is coded and decoded using both modes, and the mode requiring the smallest rate is chosen.

Unsurprisingly, results show (Figure 9) that offline mode decision performs equally well or worse than the online mode decision for skip mode, for reasons explained above. Concerning the intra mode, a significant gap still exists between the online and offline mode decision, caused by inaccurate rate estimation. Accurate rate estimation, especially Wyner-Ziv rate estimation, remains an important challenge in DVC, not only for the purpose of accurate mode decision but also for other purposes such as rate control or feedback channel free DVC systems. It is very closely related to another significant aspect in DVC, namely virtual noise estimation.

Table 7 shows the number of blocks in each mode when the offline mode decision is applied. Less blocks are skipped in this case compared to the online mode decision, which confirms that in the latter case some skipped blocks still contain errors. Opposite to that, fewer intra blocks are chosen by the online mode decision than by the offline mode decision, indicating an average underestimation of the Wyner-Ziv rate. The reason for this underestimation lies in the increased number of skipped modes. Since the Wyner-Ziv rate estimation is partly based on the Wyner-Ziv rate spent in previous frames, the occurrence of skipped blocks still containing errors causes a slight underestimation of the Wyner-Ziv rate.

5.5 Comparison with H.264/AVC and Blast-DVC

Our system (having intra, WZ, and skip mode enabled) is additionally compared to the current state-of-the-art in conventional video compression, i.e., H.264/AVC. Two configurations of the latter are considered, namely, intra coding only and inter coding. To allow a meaningful comparison, H.264/AVC has been restricted to a fixed GOP of size 4 (hierarchical coding, using only two reference frames). The extended profile was used, one slice per picture.

The results in Figure 10 indicate that our system, unlike DISCOVER, is able to outperform H.264/AVC intra coding consistently, also for sequences with moderate to high motion (such as Foreman and Table Tennis).

We also compare our system to the Blast DVC codec, for which binaries can be found online⁷. Due to the limitations of this software, tests had to be conducted for QCIF resolution and a GOP of size 2. The test sequences used have a temporal resolution of 15 frames per second. The results in Figure 11 illustrate that the block-based

⁷ <http://enpub.fulton.asu.edu/ivu/Software/DVC/BLASTDVC/BLAST.htm> (accessed December 1st, 2010)

system proposed in this paper outperforms both Blast and DISCOVER for sequences with moderate to high motion (such as Foreman and Soccer). For sequences with low motion (such as Hall Monitor), our results are better than DISCOVER and comparable to Blast. These results illustrate the effectiveness of the techniques proposed in this paper.

5.6 Feedback channel rate

To conclude this section we briefly consider the feedback channel rate. In the proposed codec, both the encoded modes and the parity bit requests need to be transmitted from the decoder to the encoder over the feedback channel. Since this concerns a different communication channel (or at least an opposite direction) than used for the actual transmission of the video data, as commonly done in DVC, this rate is not included in the rate-distortion figures shown in this section. However, compared to the actual rate spent on coding the frames, the feedback channel rate is indeed very small. Table 8 provides results for the feedback channel rate, which varies between 9 and 48 kbps. This lies between 1 and 12 percent of the actual rate.

6 Conclusions and Future Work

This paper proposed a block-based distributed video codec with decoder driven mode decision. Three coding modes are proposed: Wyner-Ziv, skip and intra. Skip blocks are selected based on a threshold on the mean squared error between reference blocks. For intra versus Wyner-Ziv blocks, mode decision is performed on a rate-distortion basis, by assuring equal distortion and by selecting the mode that requires the smallest estimated rate.

The block-based design has a major advantage over plane-based DVC codecs, namely its ability to easily adapt to the spatially varying characteristics in a video sequence. In particular, not only can the rate vary, but also the coding mode can be changed from block to block. A disadvantage of the block-based codec is that it becomes less straightforward to exploit the varying statistics of the frequency bands. An ad hoc approach to counter this limitation has been proposed. Presumably, more efficient techniques could be developed in the future.

Introducing skip and intra modes greatly improves the coding efficiency of the block-based codec. Skip blocks are mainly beneficial factor low rates and for sequences containing low motion. At higher rates and for sequences with more motion, the coding gain is mainly attributed to the use of intra blocks. Depending on the sequence, the skip and intra modes introduce an average bitrate gain of up to 33.7% over the basic block-based codec employing the Wyner-Ziv mode only, and up to 29.7% over the state-of-the-art DISCOVER codec of [2].

Rate estimation, especially Wyner-Ziv rate estimation, remains a difficult challenge. Comparing the coding performance achieved using online rate estimation with perfect, offline rate estimators shows that there would still be room for significant rate-distortion improvements if more accurate rate estimators could be developed.

Acknowledgements The research activities that have been described in this paper were funded by Ghent University, the Interdisciplinary Institute for Broadband Technology (IBBT),

the Institute for the Promotion of Innovation by Science and Technology in Flanders (IWT-Flanders), the Fund for Scientific Research-Flanders (FWO-Flanders), and the European Union.

References

1. A. Aaron, S. Rane, E. Setton, and Bernd. Transform-domain Wyner-Ziv codec for video. In *Proc. SPIE Visual Communications and Image Processing*, January 2004.
2. X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Oualet. The DISCOVER codec: Architecture, techniques and evaluation. In *Proc. Picture Coding Symposium (PCS)*, November 2007.
3. J. Ascenso, C. Brites, and F. Pereira. Improving frame interpolation with spatial motion smoothing for pixel domain distributed video coding. In *Proc. 5th EURASIP Conference on Speech and Image Processing, Multimedia Communications and Services*, July 2005.
4. J. Ascenso and F. Pereira. Low complexity intra mode selection for efficient distributed video coding. In *Proc. International Conference on Multimedia and Expo (ICME)*, July 2009.
5. Z. Belkoura and T. Sikora. Improving Wyner-Ziv video coding by block-based distortion estimation. In *Proc. European Signal Processing Conference*, September 2006.
6. S. Benierbah and M. Khamadja. Hybrid Wyner-Ziv and intra video coding with partial matching motion estimation at the decoder. In *Proc. IEEE International Conference on Image Processing (ICIP)*, November 2009.
7. W.-J. Chien and L. Karam. BLAST-DVC: BitpLane SelecTive distributed video coding. *Multimedia Tools and Applications*, 48(3):437–456, July 2010.
8. W.-J. Chien, L. Karam, and G. Abousleman. Block-adaptive wyner-ziv coding for transform-domain distributed video coding. In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2007.
9. T. Do, H. J. Shim, and B. Jeon. Motion linearity based skip decision for Wyner-Ziv coding. In *Proc. International Conference on Computer Science and Information Technology*, August 2009.
10. G. Esmaili and P. Cosman. Low complexity spatio-temporal key frame encoding for Wyner-Ziv video coding. In *Proc. Data Compression Conference (DCC)*, March 2009.
11. Y. Feng, Y. Li, C. Wu, and R. Song. Coding scheme with skip mode based on motion filed detection for dvc. In *Proc. Satellite Data Compression, Communication, and Processing IV*, September 2008.
12. B. Girod, A. Aaron, S. Rane, and D. Rebollo-Monedero. Distributed video coding. *Proceedings of the IEEE*, 93(1):71–83, January 2005.
13. M. Jakubowski. Constant rate control algorithm for wyner-ziv video codec. In *Proc. Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2009*, May 2009.
14. D. Kubasov, K. Lajnef, and C. Guillemot. A hybrid encoder/decoder rate control for a Wyner-Ziv video codec with a feedback channel. In *Proc. IEEE MultiMedia Signal Processing Workshop*, October 2007.
15. S. Ma, W. Gao, and Y. Lu. Rate-distortion analysis for h.264/avc video coding and its application to rate control. *IEEE Transactions on Circuits and Systems for Video Technology*, 15(12):1533–1544, December 2005.
16. S. Mys, J. Slowack, J. Škorupa, P. Lambert, and R. Van de Walle. Introducing skip mode in distributed video coding. *Signal Processing: Image Communication*, 24(3):200–213, March 2009.
17. F. Pereira, L. Torres, C. Guillemot, T. Ebrahimi, R. Leonardi, and S. Klomp. Distributed video coding: Selecting the most promising application scenarios. *Signal Processing: Image Communication*, 23(5):339–352, June 2008.
18. R. Puri, A. Majumdar, and K. Ramchandran. PRISM: A video coding paradigm with motion estimation at the decoder. *IEEE Transactions on Image Processing*, 16(10):2436–2448, October 2007.
19. R. Puri and K. Ramchandran. PRISM: A new robust video coding architecture based on distributed compression principles. In *Proc. Allerton Conference on Communication, Control and Computing*, October 2002.

20. R. Puri and K. Ramchandran. PRISM: A “reversed” multimedia coding paradigm. In *Proc. IEEE International Conference on Image Processing (ICIP)*, September 2003.
21. D. Slepian and J. K. Wolf. Noiseless coding of correlated information sources. *IEEE Transactions on Information Theory*, 19(4):471–480, July 1973.
22. J. Slowack, S. Mys, J. Škorupa, P. Lambert, C. Grecos, and R. Van de Walle. Accounting for quantization noise in online correlation noise estimation for distributed video coding. In *Proc. Picture Coding Symposium (PCS)*, May 2009.
23. S. Sofke, F. Pereira, and E. Müller. Dynamic quality control for transform domain Wyner-Ziv video coding. *EURASIP Journal on Image and Video Processing, Special Issue: Distributed Video Coding*, pages 1–15, January 2009.
24. M. Tagliasacchi, J. Pedro, F. Pereira, and S. Tubaro. An efficient request stopping method at the turbo decoder in Distributed Video Coding. In *Proc. EURASIP European Signal Processing Conference*, September 2007.
25. M. Tagliasacchi, A. Trapanese, S. Tubaro, J. Ascenso, C. Brites, and F. Pereira. Intra mode decision based on spatio-temporal cues in pixel domain Wyner-Ziv video coding. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, May 2006.
26. A. Trapanese, M. Tagliasacchi, S. Tubaro, J. Ascenso, C. Brites, and F. Pereira. Embedding a block-based intra mode in frame-based pixel domain Wyner-Ziv video coding. In *Proc. International Workshop on Very Low Bitrate Video*, September 2005.
27. D.-C. Tsai, C.-M. Lee, and W.-N. Lie. Dynamic key block decision with spatio-temporal analysis for wyner-ziv video coding. In *Proc. IEEE International Conference on Image Processing (ICIP)*, October 2007.
28. J. Škorupa, J. Slowack, S. Mys, P. Lambert, C. Grecos, and R. Van de Walle. Stopping criterions for turbo coding in a Wyner-Ziv video codec. In *Proc. Picture Coding Symposium (PCS)*, May 2009.
29. T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra. Overview of the H.264/AVC video coding standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(7):560–576, July 2003.
30. A. D. Wyner and J. Ziv. The rate-distortion function for source coding with side information at the decoder. *IEEE Transactions on Information Theory*, 22(1):1–10, January 1976.

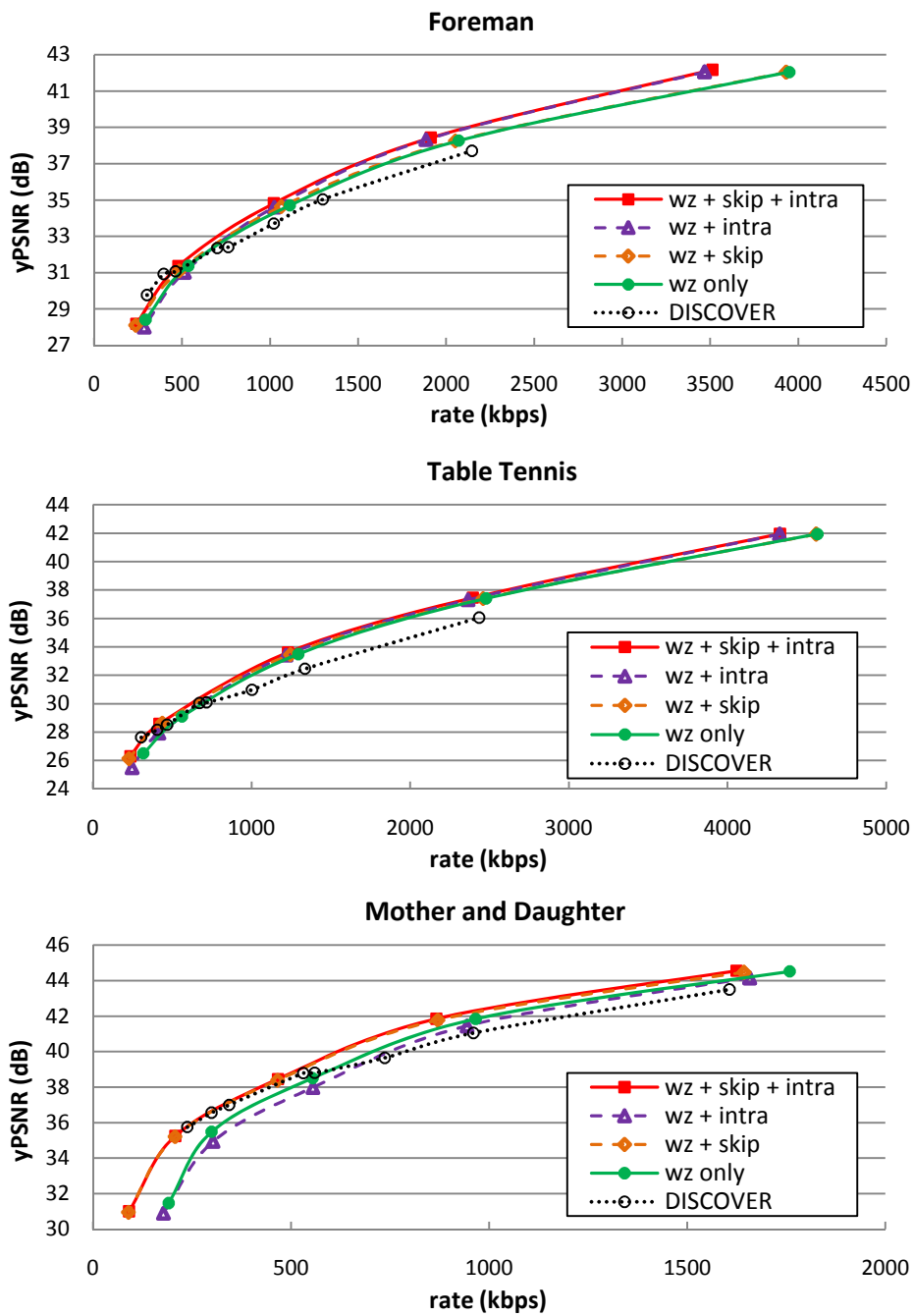


Fig. 8 Rate-distortion performance of the proposed block-based codec with decoder-driven mode decision (CIF, 30 fps, GOP 4).

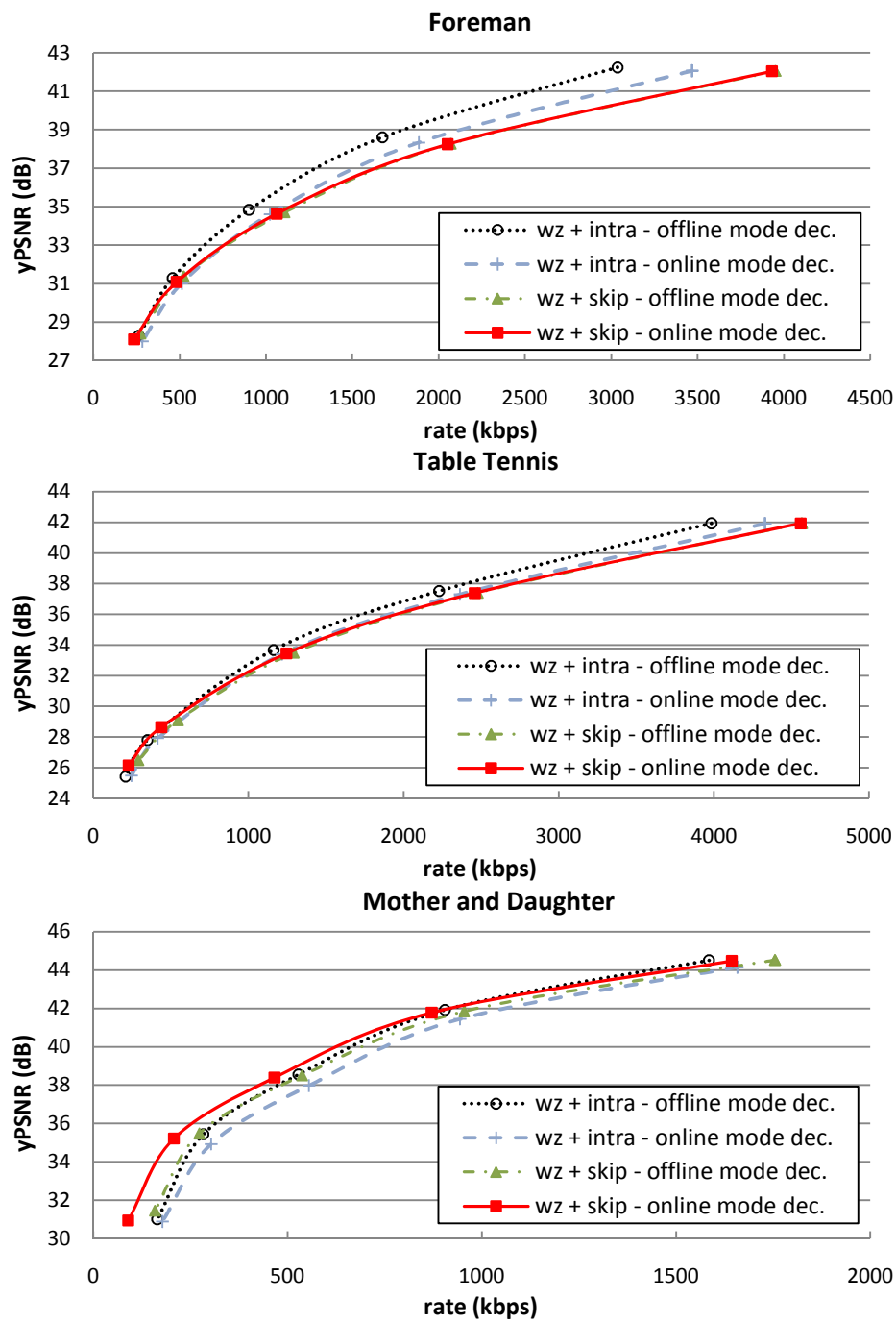


Fig. 9 Online versus offline mode decision in the proposed block-based codec.

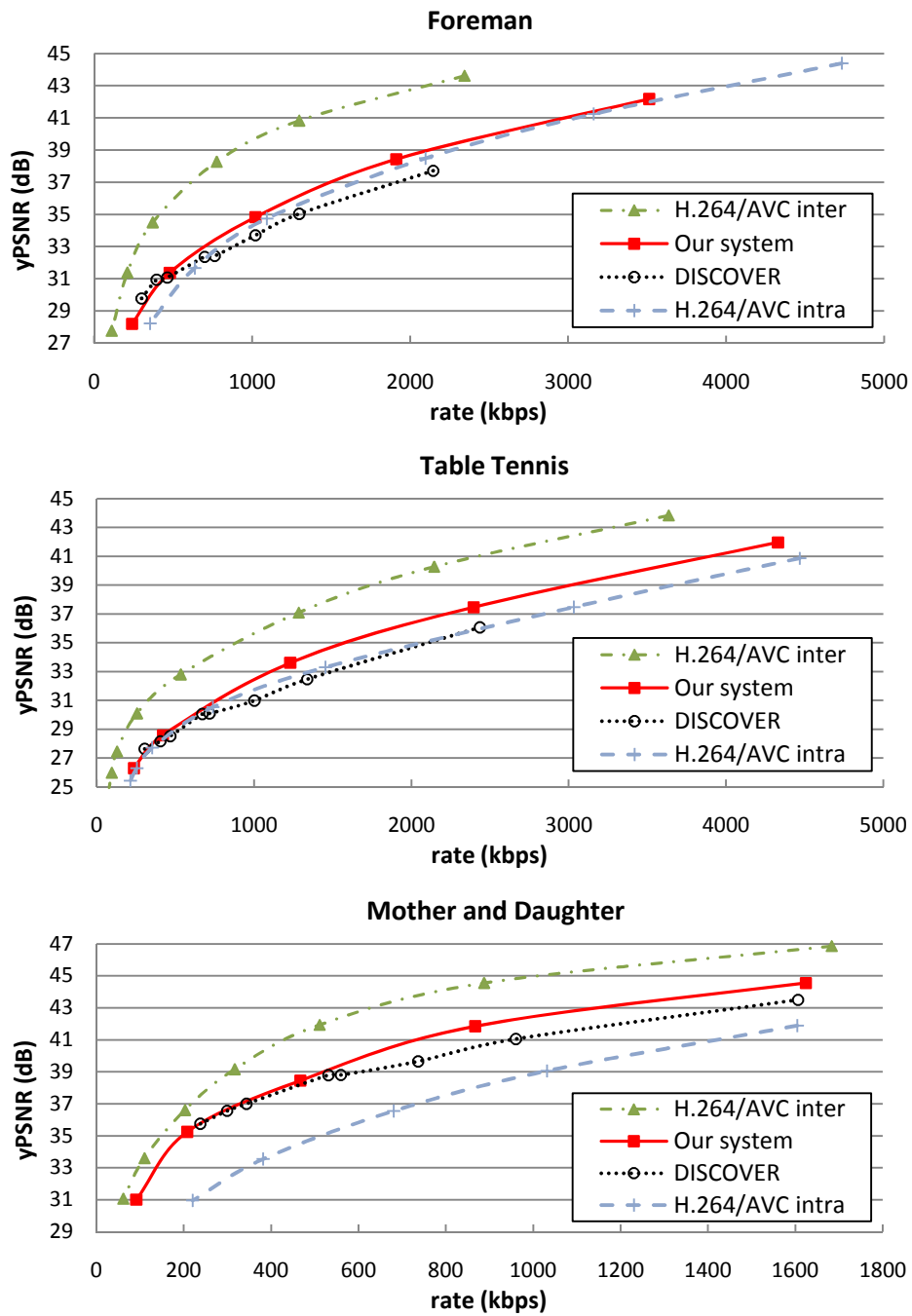


Fig. 10 Rate-distortion performance of the proposed block-based codec compared to H.264/AVC and DISCOVER (CIF, 30 fps, GOP 4).

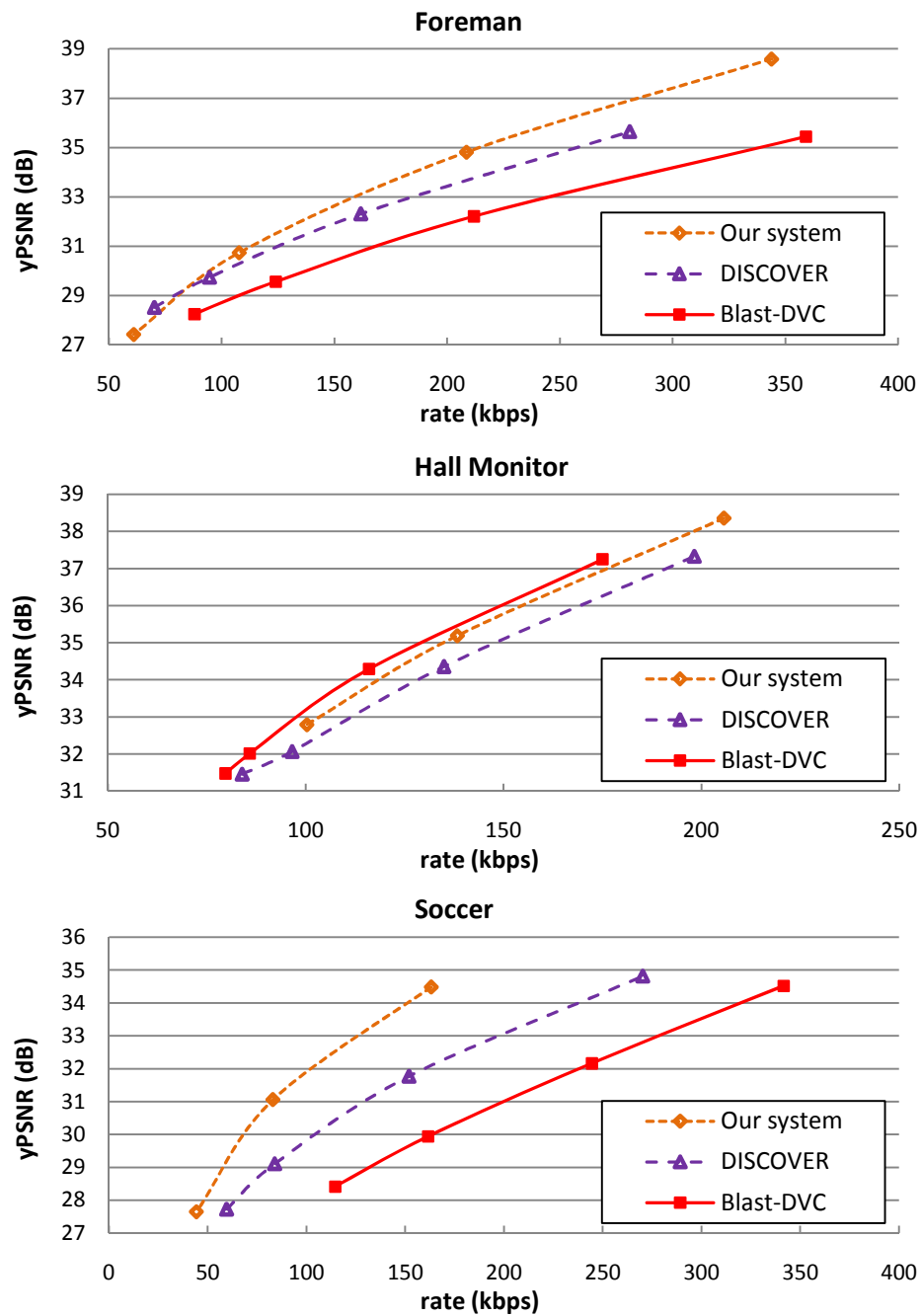


Fig. 11 Rate-distortion performance of the proposed block-based codec compared to DISCOVER and Blast (QCIF, 15 fps, GOP 2).

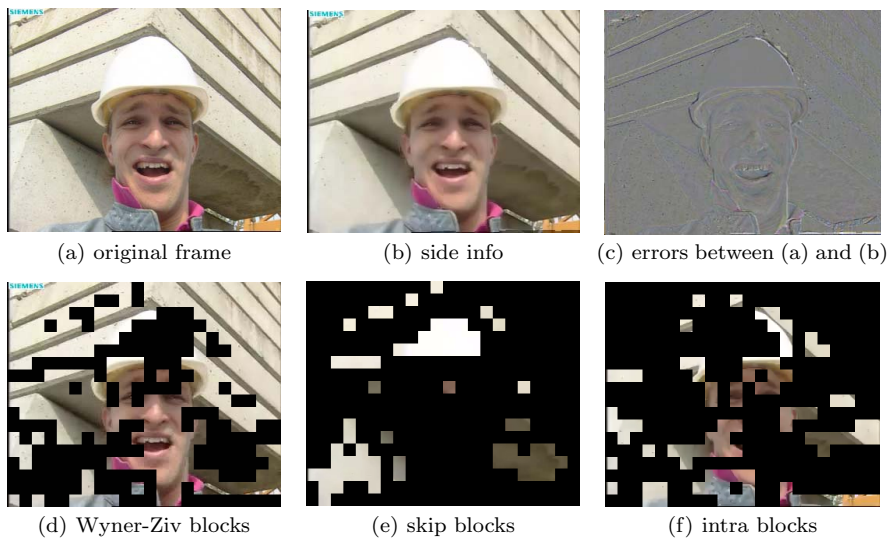


Fig. 12 Example of a Wyner-Ziv frame coded with the proposed codec using online mode decision. Foreman sequence, CIF, Q3, frame 22.