Hybrid Multiple Description Coding Based on H.264

Chia-Wei Hsiao and Wen-Jiin Tsai, *Member, IEEE*

***Abstract*—Multiple description (MD) video coding is one of the approaches that can be used to reduce the detrimental effects caused by transmission over error-prone networks. A number of approaches have been proposed for MD coding, where each provides a different tradeoff between compression efficiency and error resilience. This paper first presents two basic MD coding methods; one segments the video in the spatial domain, while the other in the frequency domain. Then a hybrid MD coding method is proposed. The hybrid MD encoder segments the video in both the spatial and frequency domains. In the case of data loss, the hybrid MD decoder takes advantage of the residual-pixel correlations in the spatial domain, and the coefficient correlations in the frequency domain, for error concealment. As a result, better error resilience can be achieved at high compression efficiency. The advantages of the proposed hybrid MD method are demonstrated in the contexts of descriptor loss in ideal channels and in packet-loss networks.**

***Index Terms*—Frequency segmentation, multiple description coding, spatial segmentation.**

I. Introduction

THROUGH the growing of communication technology, video streaming has recently become a popular field. There had been more and more application services about video streaming developed and provided, such as IPTV, peer-to-peer (P2P) live video, and video phone; the scale of these services also becomes larger. Transmitting video streams smoothly to effectively combat network errors is an important subject.

H.264/AVC is a video coding standard developed by Joint Video Team, founded by ITU-T and ISO/IEC. It has a better video quality and compression efficiency than most of the existing standards, such as MPEG2 and H.263. When transmitting the H.264/AVC encoded bit-stream, as the coding efficiency is high, the bits of the stream carry more information of the video source and are more vulnerable to transmission errors. Therefore, there had been a lot of error resilience tools proposed to combat transmission errors. Low-bandwidth hand-held devices have become more popular and backbone capacities of the Internet have increased, thus for a video streaming service, the client bandwidth varies in a wide range, from hundreds of kilo-bytes to tens of mega-bytes. Clients on hand-held devices such as cell phones, smart phones, or PDAs usually have lower bandwidth, while in desktop computers higher bandwidth is common. As a result, a coding technique that enables the services to be adaptive to the varying bandwidth of heterogeneous networks and devices would become more appealing.

MDC is a technique that encodes a single information source into two or more output streams, called *descriptors*, and each descriptor can be decoded independently and has an acceptable decoding quality; in addition, the decoding quality will be better if more descriptors were received. Fig. 1 shows the conventional MDC system architecture. The encoder encodes the source into two individual descriptors and then sends through two channels. The decoder has multiple decoder states: *side decoder* and *center decoder*; when receiving only one descriptor, the side decoder is responsible for decoding the one-descriptor bit-stream; if both descriptors are received, the center decoder is used to produce the best quality output. Contrary to MDC, we refer to single description coding (SDC) as the standard H.264/AVC bit-stream.

The first MD video coder, called multiple description scalar quantizer (MDSQ) [4], has been realized in 1993 by Vaishampayan, who proposed an index assignment table that maps a quantized coefficient into two indices each could be coded with fewer bits. Afterward, research on different 66 MDC approaches had been proposed. These approaches can be intuitively classified through the stage where it split the signal, such as spatial domain, frequency domain, and temporal domain. To be more precise, Wang [3] came up with another classification scheme which was based on the type of predictor a MDC approach adopted, and three classes were defined. Class A focuses on prediction efficiency; class B focuses on mismatch control; and class C controls tradeoff between the two issues.

Class A model applies the MDC after motion compensation and has the property that the predictor used in the encoder is in accordance with that used in the center decoder. In other words, there is only one prediction loop for motion estimation, and the reference frames used in the encoder must be fully reconstructed as if all descriptors were received in the decoder. As a result, class A encoder has high prediction efficiency because its prediction is the same as that in SDC. There are a number of MDC approaches using the model of class A. These approaches split the signal either on frequency

Manuscript received November 12, 2008; revised March 14, 2009. First version published July 7, 2009; current version published January 7, 2010. This paper was recommended by Associate Editor D. S. Turaga.

C.-W. Hsiao is with Alpha Image Technology Corporation, Jubei City, Hsinchu 302, Taiwan (e-mail: ultraxiao@gmail.com).

W.-J. Tsai is with the Department of Computer Science, National Chiao-Tung University, Hsinchu 300, Taiwan (e-mail: wjtsai@cs.nctu.edu.tw).

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.2009.2026973

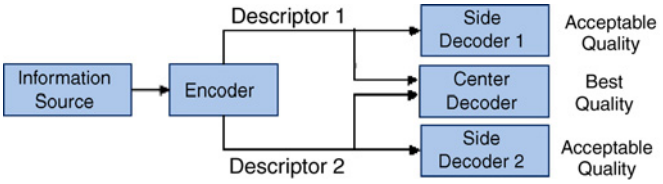


Fig. 1. Conventional MDC system architecture.

coefficients or on residual data. The first implementation of MDC, MDSQ [4], and the latter in [6] which had applied the MDSQ approach to H.264/AVC, use the architecture of class A model because there is only one prediction loop. After the prediction, the transform and quantization are performed and then the coefficients are split into two paths, generating two descriptors. In [8], the transformed coefficients are split into two descriptors such that the total distortion and bit-rate of two descriptors are minimized by Lagrange multiplier λ. Although the generated descriptors have optimal total distortion, they have no equivalent quality and bit-rates. A two-stage splitting method in [9] is proposed to combat this issue. In [9], the coefficients are first assigned to obtain two energy equivalent descriptors, resulting in balanced distortion, and then the coefficients are swapped to make sure that the two descriptors have nearly the same bit-rate. Similar approaches based on frequency coefficient splitting can also be found in [10]. Jia and Kim [11] adopted a different approach which performs the splitting on prediction error. The residual of each macroblock after motion compensation is split into two parts, then a new data partition mode is added to generate the two descriptors.

The main disadvantage of class A model is the distortion coming from the mismatch between the prediction loops at the encoder and the decoder for the loss of some descriptors. Class B model is characterized by the prediction mismatch control, which is achieved by having the prediction loop in the encoder be the same as that in the side decoder of each descriptor. In other words, MDC is applied before the motion compensation stage so the encoder can encode each descriptor separately. As a result, a better side decoder quality can be achieved in the case of descriptor loss. However, class B model suffers from the problem of prediction inefficiency because incomplete frame information is used for prediction, which either uses partial information contained in individual descriptor or uses partial information common in every descriptor. Since the predicted blocks used are not the same as those in SDC, the coding bit-rate increases for a given quality.

A variety of MDC approaches adopt class B model, from simple to complex architectures. The simplest approach might be the one that splits the video sequence into odd and even frames, separately encodes the two groups to form two descriptors, and applies the estimation of lost description in the side decoders [12]. The prediction inefficiency increases when the temporal distance increases. Therefore, if more descriptors are to be generated, the prediction for each descriptor becomes more inefficient. In [13], a more complex architecture is proposed. Two types of frames, H-SNR for high quality and L-SNR for low quality, are alternative placed in two descriptors and two-stage quantization is used. H-SNR frames

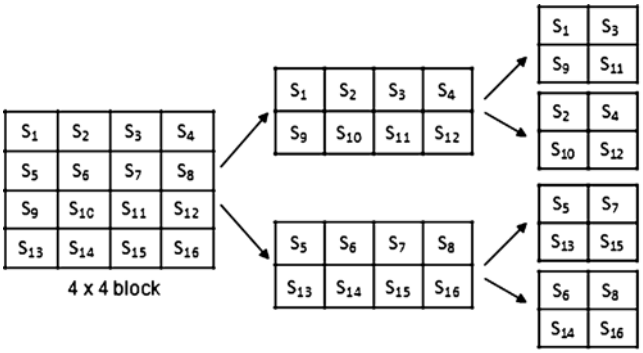


Fig. 2. Polyphase subsampling.

are produced in the first stage and L-SNR frames are produced in the second stage quantization. The mismatch control is done by using the L-SNR frames as reference frames, since H-SNR could be transformed to L-SNR for the second stage quantization in the decoder. This model is an example of class B with the type that uses information common in both descriptors for prediction. Reference [14] is another class B model based on H.264/AVC. It utilizes the slice group with disperse mode, which groups macroblocks in a frame to two slices and forms a check board pattern. In one descriptor, one of the two slices is quantized by a higher quantization parameter (QP) and the other with a lower QP, and in the other descriptor, the QP is reversed. Since lower QP has higher quality, if two descriptors are all received, the lower QP slices in each descriptor are displayed; while if only one descriptor is received, the two slices, one of higher QP and one of lower QP, in this descriptor are displayed.

The polyphase spatial subsampling (PSS) method [7] is designed for generating four descriptors. The encoder and decoder used in [7] are a conventional H.264/AVC encoder and decoder. The MDC is done before the encoder and the merging is done after the decoder. The PSS method splits each frame of the original sequence into four subframes; each has half the width and height as shown in Fig. 2, where we assume the left 4×4 block is the original frame with resolution 4×4, and it is first subsampled by factor 2 row-by-row and then column-by-column. In [7], a nonlinear interpolator, called *edge-sensing*, is proposed for the estimation of lost description in the case of receiving three descriptors, while in other cases a near neighbor replicator (NNR) and a conventional bilinear interpolator are used. The edge-sensing algorithm is based on a gradient calculation for each lost pixel in x and y directions. With the two gradients, the smoother direction can be determined, and averaging the pixels in this direction has a better error concealment effect than using a bilinear interpolator.

Most MDC methods are limited to two descriptors [17], [18]. This is a very heavy constraint for a scalable environment when more than two levels of reconstruction are required, or for high bit-rate applications where having a multilayer representation of the source is useful. Furthermore in a P2P environment where MDC is used to split video streams into multiple descriptors and peers forward these descriptors independently, it is advantageous to have more than two

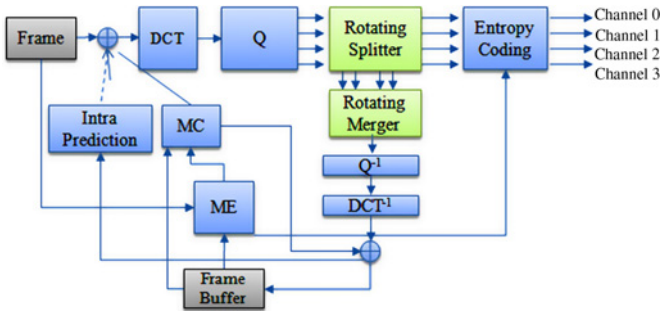


Fig. 3. Encoder architecture of D4.

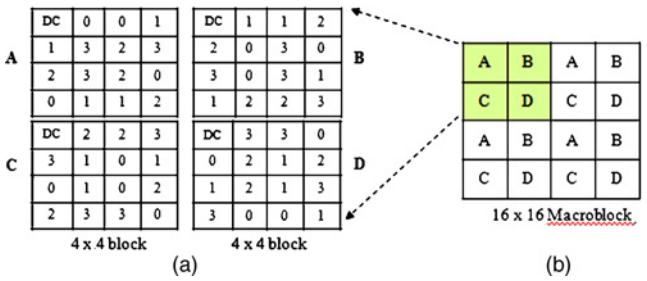


Fig. 4. Four block types and their distributions in D4. (a) Four 4 × 4-block types. (b) Block-type rotating.

descriptors to protect peers against *churn* (peers depart at a high rate) because each peer can continue viewing a video as long as it receives at least one descriptor. For the above reasons, this paper aims at generating *four* descriptors. Two basic MDC methods, called R4 and D4, and a hybrid MDC method are proposed. R4 applies MDC on the spatial domain, while D4 applies MDC on the frequency domain. The novelty of the Hybrid method is that the MDC is applied on both spatial and frequency domains such that the side decoder can take advantage of the data correlation in the two domains to have better estimation of lost description. Since the proposed methods are based on the model of class A where only one prediction loop is needed, the prediction efficiency could be as good as SDC, no matter how many descriptors are generated.

This paper is organized as follows. Section II introduces two basic MDC methods and Section III presents a novel hybrid MDC method. Experimental results are shown and discussed in Section IV. Concluding remarks are given in Section V.

II. Two Basic MDC Methods

Two basic MDC methods, called D4 and R4, are presented in this section. D4 applies MDC on frequency coefficients, while R4 on prediction error (residual data). In both methods, header information, motion vectors, and intracoded macroblocks are duplicated for every descriptor.

A. D4

Fig. 3 shows the overall encoder architecture of D4, where there is a rotating splitter block in between quantization and entropy coding blocks. The rotating splitter block splits ac coefficients into four parts and outputs into four data paths. Data from the four paths are then separately entropy encoded to generate four bit-streams, one for each descriptor. D4 is a class-A method, where there is only one prediction loop, and needs a single reference frame. To reconstruct the frame, the rotating merger block is performed on the four paths output from rotating splitter, and then the paths are merged into one for succeeding inverse quantization, inverse transform and motion estimation, just like the conventional H.264 prediction loop.

The rotating splitter performs ac coefficient splitting based on the 4 × 4 integer DCT block in H.264/AVC. For each 4 × 4 block, the rotating splitter duplicates the dc coefficient to every descriptor and splits the 15 ac coefficients in a way that alternatively assigned the coefficients in the zig-zag scanning order to the four descriptors; that is, for every four consecutive ac coefficients in zig-zag scanning order, the first coefficient is assigned to the first descriptor, the second one is assigned to the second descriptor, etc.

There is a quality-unbalanced problem in the alternative assignment described above because the first descriptor, which always carries the lowest frequency in every consecutive four coefficients will have best quality, while the fourth descriptor which carries the highest frequency will have the worst quality. To balance the quality of the four descriptors, the rotating splitter rotates the coefficient assignment among descriptors, and thus generates four types of 4 × 4 blocks: A, B, C, and D, as illustrated in Fig. 4. The number in each block indicates the descriptor number that the coefficient is assigned. As Fig. 4(a) shows, type A begins by assigning ac0 to descriptor 0; type B to descriptor 1; type C to descriptor 2; and type D to descriptor 3. The four types of blocks are equally distributed in each macroblock in order to make the resulting descriptors have balanced quality as shown in Fig. 4(b). Another gain from this type of rotating assignment is that the estimation of lost description in the decoder side can be performed efficiently.

The decoder is responsible for decoding and merging the descriptors. When two or more descriptors are received, the decoder merges the coefficients before dequantization and inverse DCT transform of a 4 × 4 block. The merge can be done by simply adding the coefficients of the same positions from different descriptors. With regard to the estimation of lost description, it is done by ac-coefficient prediction through neighboring 4 × 4 blocks. Due to the rotating block-type distribution, the colocated coefficients of neighboring blocks must belong to different descriptors and have very little chance of losing (description) simultaneously. Therefore, estimation of lost description is efficient.

Fig. 5 shows the example for the estimation of a lost 8 × 8 block, where we assume descriptor 3 is lost and the positions labeled "X" mean the corresponding ac coefficients assigned to descriptor 3. As we can see, the left-top block is type A, and the lost coefficients in this block are colocated to the coefficients assigned to descriptor 0 in the right-top block of type B. Since descriptor 0 is not lost, the three coefficients in the type B block can be copied to the type A block. In a similar way, the coefficients belonging to descriptor 1 in the

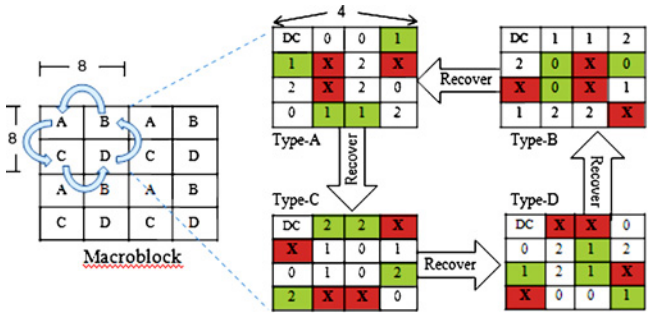


Fig. 5. Coefficient prediction for lost descriptor 3.

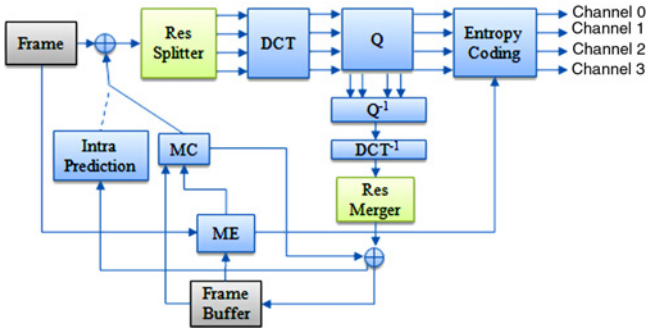


Fig. 6. Encoder architecture of R4.

bottom-right type-D block are used to recover the coefficients labeled "X" in the type B block, etc. The estimation directions are indicated in the figures by four arrows.

B. R4

Fig. 6 shows the encoder architecture of another basic MDC method, called R4, where there is a res-splitter block to split each macroblock's residual data into four macroblocks, one for each descriptor. As can be seen, the major difference between R4 and D4 is that one performs the splitting before DCT, but the other does it after quantization. R4 is also a class-A method. The reconstruction of a single reference frame is done in a similar way to D4, and thus is not addressed here again.

Fig. 7 shows how the res-splitter in R4 splits a macroblock. The residual data in a macroblock are divided into four 8×8 blocks; each of them is assigned to one descriptor. Since each descriptor obtains only one-quarter of the original residual data, there will be three 8×8 blocks which are not assigned with any residual data in each descriptor. The DCT coefficients of these 8×8 blocks are set to all zero, thus the coded block pattern (CBP) will not be set and the bits to encode each descriptor can be saved. The estimation of lost description in R4 is done in the residual domain by using the prediction of residual data from the neighboring 8×8 blocks. For a lost 8×8 block, it is reconstructed by filling all its lost residual value with x, which is the mean of the received residual values in the same macroblock. Since residual data in the same macroblock has spatial correlation, it is benefit to utilize this property.

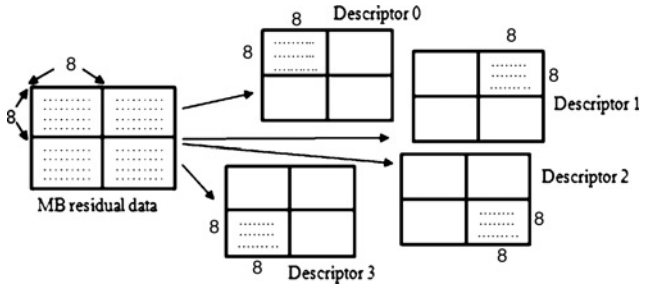


Fig. 7. Residual data splitting in R4.

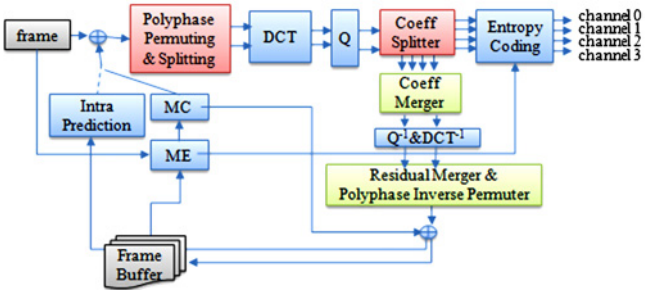


Fig. 8. Hybrid encoder architecture.

III. Hybrid Model

The hybrid model (called Hybrid) is proposed as an improved method based on the two basic ones described above. It is designed to explore both the spatial correlation between adjacent residual data and the frequency correlation between neighboring blocks. Fig. 8 shows the encoder architecture. The Hybrid encoder has a two-level splitting process in the encoding loop: 1) Polyphase Permuting and Splitting and 2) CoeffSplitter; the former splits block data in the residual domain, while the latter splits the transformed coefficients in the frequency domain. The encoding path is split into two after the first-level splitting and then four after the second level splitting. The four paths are merged before the inverse quantization in order to reconstruct full information of the reference frame, just like D4 and R4.

A. Polyphase Permuting and Splitting

The first-level splitting, Polyphase Permuting and Splitting, is applied on the 8×8 blocks in the residual domain. After motion compensation, the residual data in each 8×8 block are first polyphase permuted inside the block and then split into two blocks, as shown in Fig. 9. Before permuting, the residual pixels in the 8×8 block are first labeled with numbers ranging from 0 to 3. The labeling mechanism is that, for every 2×2 pixels, 0 is labeled on top-left pixel, 1 on top-right pixel, 2 on bottom-left pixel, and 3 on bottom-right pixel. The polyphase permuting then rearranges all the pixels of label-0 to the top-left 4×4 block, the pixels of label-1 to the top-right 4×4 block, etc., as illustrated in the middle of Fig. 9, where only one 8×8 block is shown. There are four 8×8 blocks in each macroblock. All of them are permuted in the same way. To permute pixels inside each 8×8 block before splitting is to

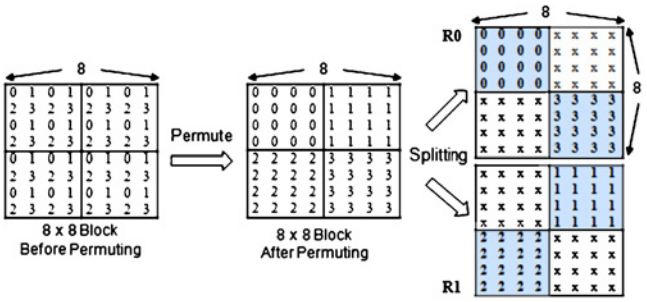


Fig. 9. Permuting and splitting of a 8 × 8 block.

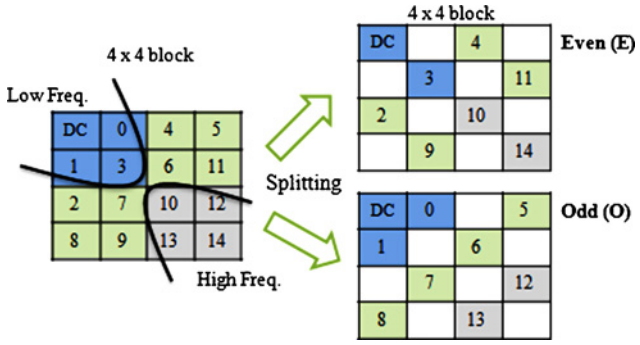


Fig. 10. Even and odd 4 × 4 blocks generated by CoeffSplitter.

take into account the estimation method of lost description, which will be discussed later in this section.

After polyphase permuting, the splitting process is performed to split each 8 × 8 block into two 8 × 8 blocks, called residual 0 (R0) and residual 1 (R1); each carries two 4 × 4 residual blocks chosen in diagonal: top-left and bottom-right 4 × 4 residual blocks are in one 8 × 8 block, while top-right and bottom-left ones are in the other 8×8 block. For each 8×8 block, the remaining two 4 × 4 blocks with pixels all labeled with "x" in Fig. 9 are given residual pixels all set to zero. The encoder has no need to encode the coefficient of these two all-zero 4 × 4 blocks. As Fig. 8 shows, the encoding path is split into two after the first-level splitting. For each path, DCT is performed to every 4 × 4 block (except those all-zero blocks), and then the second level splitting, CoeffSplitter, is applied.

B. CoeffSplitter

The second level splitting, CoeffSplitter, is based on the splitting of the DCT ac coefficients in the frequency domain. It is modified from the D4 method described in Section II-A. In D4, the coefficients are assigned to each of the four descriptors alternatively, which may result in unbalanced qualities among descriptors; while in hybrid, the coefficient splitting is modified to improve this drawback.

It is known that DCT coefficients have different importance from the viewpoints of human's subjective visual quality. The coefficients of lower frequency are more important because they are more sensitive to the human visual system, while coefficients of higher frequency are generally less important. The CoeffSplitter takes into account the different importance of DCT coefficients and divides the 16 DCT coefficients of a 4 × 4 block into three groups: 1) low frequency; 2) median frequency; and 3) high frequency, as shown in Fig. 10(a): the four coefficients closest to dc are assigned to the low frequency group; the four furthest from dc are assigned to the high frequency group, and others are to the median frequency group. Based upon this grouping strategy, CoeffSplitter splits the ac coefficients of each group into two parts, one for each descriptor. dc is duplicated because it is the most important coefficient.

Fig. 10(b) shows how a 4 × 4 block is split into two 4 × 4 blocks; each carries almost half the total number of original coefficients. The 4 × 4 block which carries even numbers of original ac coefficients is called an *E-block*, while the other is an *O-block*. Besides the dc which is duplicated for

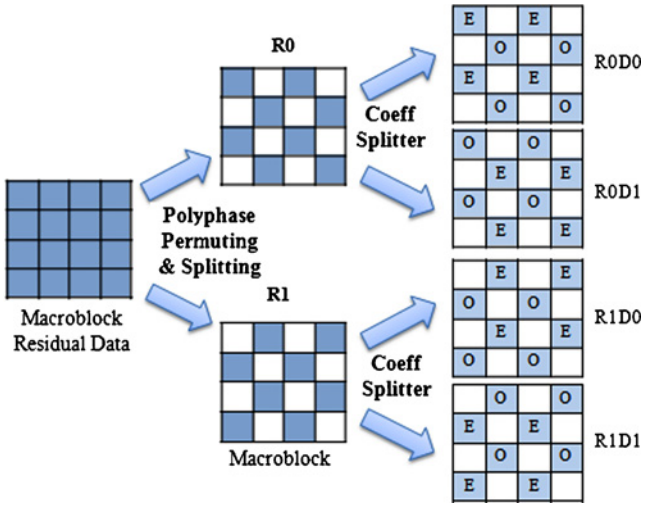


Fig. 11. Macroblock pattern after two-level splitting.

both blocks, each ac group is divided in the diagonal direction to achieve a balanced visual quality. As the figure shows, every other top-right to bottom-left diagonal of coefficients are assigned to the same descriptor, resulting in a reduced number of (Run, Level) pairs for each descriptor and therefore entropy encoding, such as CABAC or CAVLC, will be more effective. The E-blocks and O-blocks are assigned to two descriptors in an alternative diagonal pattern, as illustrated in Fig. 11.

In Fig. 11, R0 and R1 are the macroblocks split from the first-level splitting. The white color blocks in R0 and R1 are all-zero 4 × 4 blocks. The second-level splitting, CoeffSplitter, is applied on R0 and R1 to split each nonzero 4 × 4 block into O-blocks and E-blocks, which are then alternately assigned to two different macroblocks, called D0 and D1. The D0 and D1 coming from R0 are called *R0D0* and *R0D1*, respectively, and those from R1 are called *R1D0* and *R1D1*. The purpose of assigning even and odd blocks in an alternate diagonal pattern in the resulting macroblock is for better estimation of lost description, as will be discussed later. In addition, this pattern balances the quality difference as well as bit-rate difference between E and O-blocks because O-blocks have one more coefficient than E-blocks.

As a result of two-level splitting, every residual macroblock is split into four macroblocks, one for each descriptor.

TABLE I
BIT-RATES OF CAVLC CODED TEXTURE DATA WITH/WITHOUT COEFFSPLITTER

	Carphone	*Coastguard*	*Foreman*	*Mobile*	*News*
Without CoeffSplitter (kb/s)	107.831	245.964	146.236	451.55	96.63
Without CoeffSplitter (kb/s)	167.765	295.313	227.445	581.067	186.464
Redundancy	55.58%	20.06%	55.53%	28.68%	92.97%

It is worth mentioning that, after CoeffSplitter is applied, each 4×4 block (obtained from the first-level splitting) is split into two 4×4 blocks, each carrying almost half the number of the original coefficients. For those coefficients which are assigned to one block, they will be set to zero in the other block, and therefore, almost half the number of coefficients are zero in each resulting 4×4 block. Although entropy coding such as CABAC or CAVLC is still applicable for the resulting blocks, the coding efficiency is affected. Table I shows the redundancy in the bit-rate caused by CoeffSplitter for five different video sequences with QP = 28; note that only the entropy coded texture bits are counted (i.e., the bits used for motion vectors, block modes, etc. are not included) and CAVLC is used. As the results indicate, the redundancy ranges from 20% to 93%, with low-motion sequences such as *News* at the higher end. This is mainly due to the fact that dc coefficients are large in low-motion sequences and they are duplicated for each resulting descriptor after CoeffSplitter.

C. Frequency and Spatial Merging

The same to R4 and D4, the Hybrid method adopts class-A model, which uses a single prediction loop. To construct the full information of the reference frame, a two-level merging is used as illustrated in Fig. 8, where the CoeffMerger is applied in frequency domain by merging R0D0 and R0D1, and merging R1D0 and R1D1. The merging is performed by adding ac coefficients in the same positions of even and odd blocks and choosing one of the two duplicated dc coefficients. After inverse transformed, R0 and R1 are obtained and then the second level merging is applied based on 8×8 blocks. The *Residual Merger* is done first by discarding the all-zero 4×4 blocks and then *Polyphase Inverse Permuting* is performed to reconstruct the original 8×8 blocks. For each macroblock, the four 8×8 blocks are all processed in this way.

D. Hybrid Decoder

The Hybrid decoder architecture is shown in Fig. 12. The four descriptors are labeled with R0D0, R0D1, R1D0, and R1D1. These descriptors are first entropy decoded separately, then "CoeffMerger" and "Residual Merge and Polyphase Inverse Permuting" are performed in the same way on the encoder side.

If the decoder does not receive all the descriptors intact, then either frequency or spatial estimation method of lost description is adopted to reconstruct the lost data. Table II summarizes the cases for spatial or frequency estimation to be applied, where F denotes the *frequency estimation*, and S the *spatial estimation*. The columns describe the four possible cases for the two descriptors split from R0; while the rows describe those for R1. As can be seen from the table, spatial

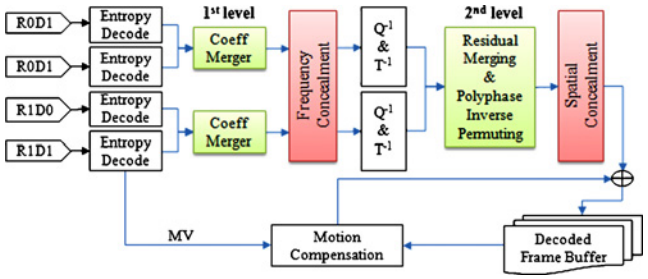


Fig. 12. Hybrid decoder architecture.

TABLE II
SUMMARY OF SPATIAL AND FREQUENCY ESTIMATION CASES

Estimation methods		Descriptor(s) in **R0**			
		D0 + D1	D0	D1	Loss
Descriptor(s) in **R1**	D0 + D1	N/A	F	F	S
	D0	F	F	F	S
	D1	F	F	F	S
	Loss	S	S	S	N/A

S: Spatial Estimation. F: Frequency Estimation.

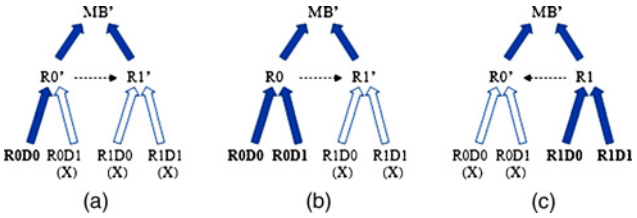


Fig. 13. Spatial estimation of lost description. (a) Three-descriptor loss. (b) Two-descriptor loss. (c) Two-descriptor loss.

estimation is applied only when one descriptor is received or two from the same residual domain (D0 + D1 from R0 or D0 + D1 from R1) are received; while for all other cases, frequency estimation is applied. The estimation algorithms for lost description are described as follows.

1) *Spatial Estimation of Lost Description:* Fig. 13 illustrates the cases where spatial estimation will be applied, where (a) illustrates one of the four possible cases that three descriptors are lost, while (b) and (c) depicted the cases of two-descriptor loss in the same residual domain. The descriptors marked with (X) mean they are lost, and R0′, R1′, and MB′ stand for the concealed version of R0, R1, and MB, respectively.

For each case in Fig. 13, only one of the R0s and R1s can be fully constructed or partially constructed from the descriptors received, and the other one is totally lost. Fig. 13(a) shows an example where R0 can be partially constructed by the received R0D0, but R1 is totally lost. Here, we propose reconstructing

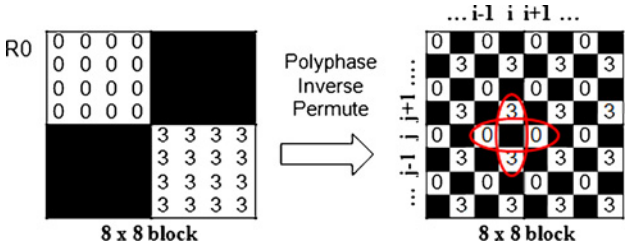


Fig. 14. Spatial estimation by bilinear interpolation.

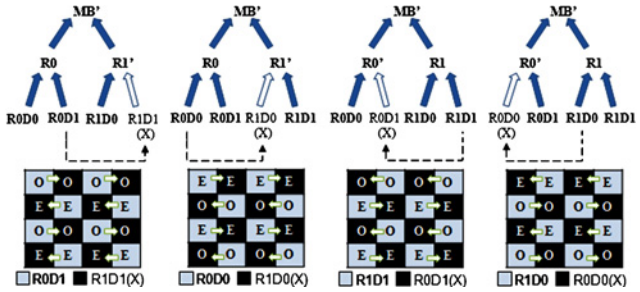


Fig. 15. ac-prediction for one missing descriptor.

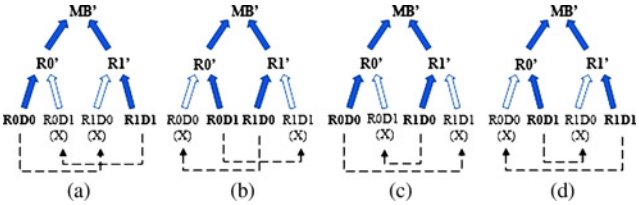


Fig. 16. ac-prediction for two missing descriptors.

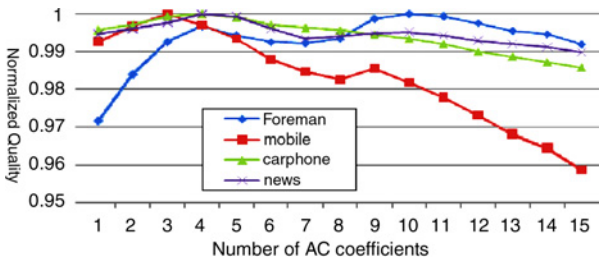


Fig. 17. Qualities by varying number of ac coefficients used in estimation.

the lost R1 by spatial estimation. After the polyphase inverse permutation of R0′, the residual pixels are distributed like a checkerboard in a macroblock as shown in Fig. 14, where for each lost residual pixel, four neighboring pixels are available. Since neighboring pixels have high spatial correlation, spatial estimation should be efficient. The spatial estimation method uses *bilinear interpolation* to reconstruct the lost residual pixels, as shown in (1) where $f_{j,i}$ is the reconstructed value of the residual pixel in column i and row j

$$\tilde{f}_{j,i} = (f_{j+1,i} + f_{j-1,i} + f_{j,i+1} + f_{j,i-1})/4. \quad (1)$$

2) *Frequency Estimation of Lost Description:* Frequency estimation is done by ac-coefficient prediction through the blocks in the counterpart of the residual domain. We call the two 8×8 residual blocks, R0 and R1, generated from the first-level splitting as the two *residual domains*. Due to polyphase permutation, two neighboring 4×4 blocks in different residual domains have one-pixel distance in the containing 8×8 block of the original image (see Fig. 11). Thus estimation of lost description through ac-prediction from the neighboring blocks in the *counterpart* of the residual domain should be efficient. Fig. 15 depicts four cases of one-descriptor loss where frequency estimation will be applied. The descriptors used for ac-prediction are from the counterpart of the residual domain as indicated by the dotted arrows. For example, in the case of R1D1 loss, the received R0D1 is used for ac-prediction.

Besides that, since E and O-blocks contain complementary coefficients, the prediction follows the principle that E-blocks are predicted from E-blocks, and O-blocks are from O-blocks. For the same example that R1D1 is lost, the lost O-blocks in R1D1 are predicted from the left-neighboring O-blocks in R0D1; while the lost E-blocks in R1D1 are predicted from the right-neighboring E-block in R0D1. As depicted in Fig. 15, the predictions for the four cases are all in a horizontal direction. After prediction for the lost descriptor, the lost residual domain can be reconstructed and then the MB is obtained.

Fig. 16 shows four out of six possible cases that two descriptors are lost. Frequency estimation is used for them, and the descriptors used for prediction are also from the counterpart of the residual domain, as the dotted arrows indicate. (*Note:* The other two cases that two lost descriptors belonging to the same residual domains adopt spatial estimation as have been described in the previous subsection.) With the design of the second-level splitting in the Hybrid encoder, the diagonal 4×4 blocks in each 8×8 block have different types (O or E-type), resulting in two directions of ac-prediction: horizontal and vertical. For cases (a) and (b) where the lost descriptors are in different residual domains and different frequency domains, the ac-predictions are in the horizontal direction. For cases (c) and (d) where the lost descriptors are in the same frequency domain, the predictions are in the vertical direction.

The frequency estimation methods described above are all based on ac-prediction from selected blocks. However, the effect of ac-prediction varies with the number of coefficients used for prediction. Fig. 17 shows the experimental results for the quality of different sequences by varying the numbers of ac coefficients used. The experiments are based on the cases of two-descriptor loss with the frequency estimation applied. The x-axis in the figure is the number of ac coefficients used for estimation. It ranges from 1 to 15 because a 4×4 integer DCT block has 15 ac coefficients. The y-axis is the normalized quality relative to the best PSNR among the 15 cases. The ac coefficients used for estimation are in the zig-zag order; that is, the value k in the x-axis means that, for every lost 4×4 block, only the first k ac coefficients in zig-zag order are recovered by copying from the predicted blocks. From Fig. 17, it is observed that all the peaks of quality fall in the interval $[3, 5]$ in most sequences. Although there are two local maximums in the *Foreman* sequence, one of them also falls in the interval $[3, 5]$. Therefore, we propose using four ac coefficients in the frequency estimation method.

IV. Experimental Results

In this section, the experimental results of the four MDC methods, PSS [7], D4, R4, and Hybrid are presented, and five test sequences: *Foreman*, *Mobile*, *Coastguard*, *Carphone*, and *News*, with QCIF (176 × 144) resolution, are used for performance evaluation. These methods are implemented by modifying H.264/AVC reference software, JM 13.2 [15]. The group of picture (GOP) size is 20 frames. The structure of each GOP is IPPPP…, the frame rate is set to 30 Hz, and the symbol mode is CABAC. The performance is measured by the reconstruction qualities of 1, 2 and 3 descriptors, the frame-by-frame quality comparison, and the qualities in packet-loss environments.

A. Side Reconstruction Performance

This section examines the performance of the proposed MDC methods in an ideal channel environment. The assumption is that some descriptors are received without losing any information while the others are totally lost. Such a situation is referred to as *side reconstruction*. Side reconstruction performance is examined for one, two, and three missing descriptors, separately. To have a fair comparison, all methods encoded streams with the same average bit-rate, 100 kb/s per descriptor.

1) *One Missing Descriptor:* In Fig. 18(a)–(c), the performance results of one-descriptor loss are presented by showing the reconstructed PSNR of three sequences with varying bit-rates. Since there are four possible cases of one-descriptor loss, the plotted PSNR is the average of them. It is observed that in Fig. 18(a), (b), Hybrid performed better than all others at all bit-rates. The PSNR gaps range from 1 to 2 dB. However, at high bit-rate in Fig. 18(c), PSS outperformed Hybrid and the performance gaps between Hybrid and the others are reduced to only about 0.5 dB. This is due to that there are many new objects appearing in the sequence of Fig. 18(c), resulting in more intracoded blocks and therefore, high bit-rates. The estimation of lost description in PSS used an edge-sensing algorithm which was effective for the content of Fig. 18(c), the *Coastguard* sequence, which has many edges in the form of horizontal coastline, ships, and waves.

2) *Two Missing Descriptors:* To illustrate the results of two-descriptor loss, Fig. 18(d)–(f) shows the reconstructed PSNR of three sequences with varying bit-rates. The plotted PSNR is the average of the six possible cases of two-descriptor loss. In these figures, it is interesting to observe that D4 outperformed R4 in *Foreman*; R4 outperformed D4 in *Coastguard*; and both R4 and D4 had similar performance in *Carphone*. This implies that neither the spatial estimation method of R4 nor the frequency estimation of D4 is suitable for all video sequences. In Fig. 18(d)–(f), Hybrid method performed better than all others for all sequences because both spatial and frequency estimation of lost description were applied for two-descriptor loss and the results reveal their effects. Compared with other methods, PSS performance degraded dramatically more in two-descriptor loss than in one-descriptor loss. This is because PSS adopted a near-neighbor replicator (NRR) instead of an edge-sensing algorithm for the case of two-descriptor

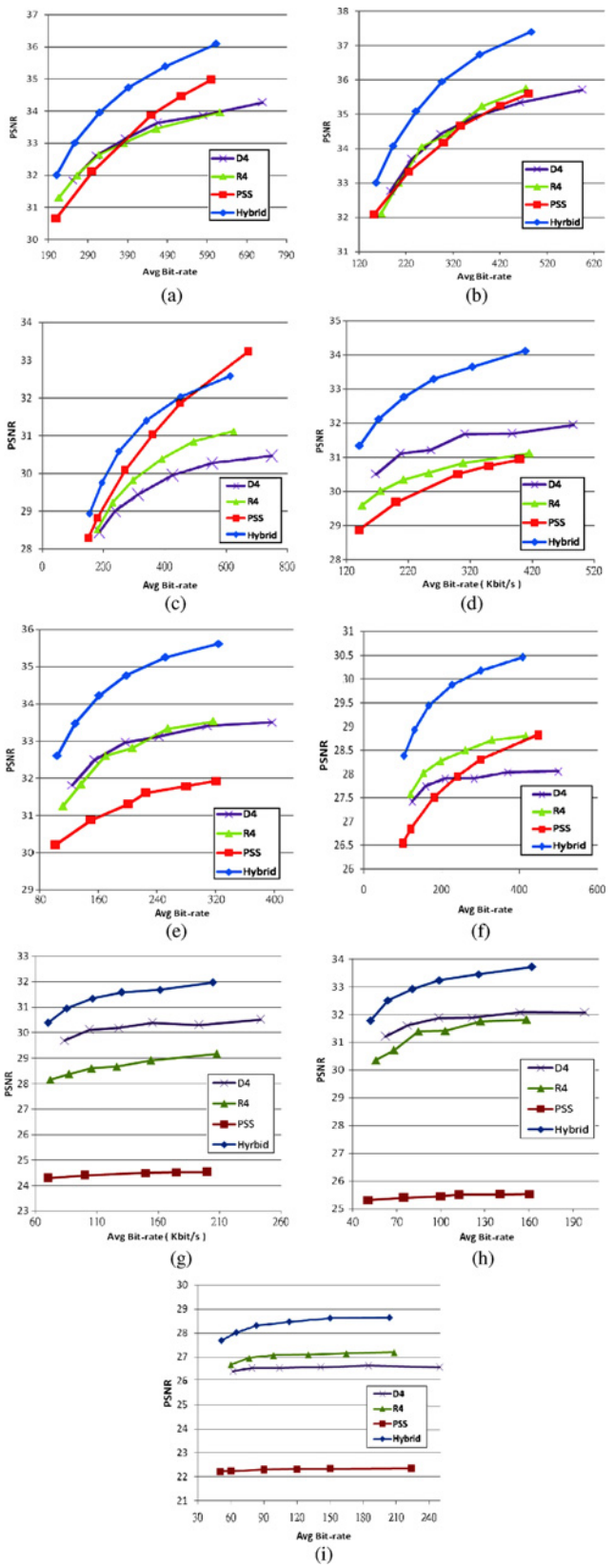


Fig. 18. PSNR of one, two, and three-descriptor loss for different bit-rates. (a) *Foreman* (one-descriptor loss). (b) *Carphone* (one-descriptor loss). (c) *Coastguard* (one-descriptor loss). (d) *Foreman* (two-descriptor loss). (e) *Carphone* (two-descriptor loss). (f) *Coastguard* (two-descriptor loss). (g) *Foreman* (three-descriptor loss). (h) *Carphone* (three-descriptor loss). (i) *Coastguard* (three-descriptor loss).

loss. The edge-sensing algorithm is more powerful, but it is only used for the case of one-descriptor loss.

3) *Three Missing Descriptors:* Fig. 18(g)–(i) presents the performance results of three-descriptor loss for three different sequences. The plotted PSNR is the average of four possible cases of three-descriptor loss. In these figures, it is observed that all the curves are more horizontal than those in Fig. 18(a)–(f). This means that the increase in bit-rate has limited effects for obtaining higher PSNR if three-fourths of the information is lost. The curves in Fig. 18(g)–(i) separate widely, with the Hybrid performing best and PSS the worst. Since the effects of estimation of lost description increase if more information is lost, the results demonstrate the superiority of the estimation approaches used in the Hybrid method.

To sum up, the experiments in Fig. 18 simulate the different demands in a heterogeneous network, where one, two, or three descriptors are needed for different devices with different capabilities. The overall results show that the proposed Hybrid method is adaptive to heterogeneous environments.

B. Frame-by-Frame Comparison

In this section, a frame-by-frame PSNR comparison of different MDC methods for missing one, two, and three descriptors is examined and the first five GOPs (100 frames) of *Carphone* sequence are shown. All methods encoded the stream with the same average bit-rate, 100 kb/s per descriptor.

Fig. 19 shows the results for (a) one-descriptor loss, (b) two-descriptor loss, and (c) three-descriptor loss. It is observed that D4, R4, and Hybrid methods have periodical quality degradation. This is because that in D4, R4, and Hybrid methods, the intracoded macroblocks are duplicated for each of the four descriptors, thus intraframes can be reconstructed with full quality even when there are some missing descriptors. However, for other frames inside the GOP, descriptor loss causes the quality degradation, and the reference-mismatch between the encoder and the side decoder causes the degradation to be propagated to the end of the GOP. Among the three methods (all use class-A model), Hybrid method has a more gradual degradation than the others. For PSS, since it is based on class-B model which has mismatch control, it has no error propagation inside a GOP as shown in Fig. 19. However, due to poor coding efficiency in class-B model, the overall PSNR of PSS is lower than that of other methods for most frames, given the same average bit-rate.

The Hybrid method yields 0.65 to 1.77 dB enhancement over other methods in Fig. 19(a), 1.4 to 3.4 dB in (b), and 1.19 to 7.77 dB in (c), showing that the more missing descriptors occur, the more benefits can be achieved by the Hybrid method. It also shows the superiority of adopting both spatial and frequency estimation methods in the Hybrid approach. There is not much difference between the performance of D4 and R4. R4 outperformed D4 for more frames in (a) and (b), while D4 outperformed R4 in (c). The performance difference between R4 and D4 depended on the types of video content. PSS suffered much worse performance in (c) than in (a) because the edge-sensing algorithm used in (a) for estimation of lost description cannot take effect in (c), which adopted NRR for reconstructing the missing three descriptors.

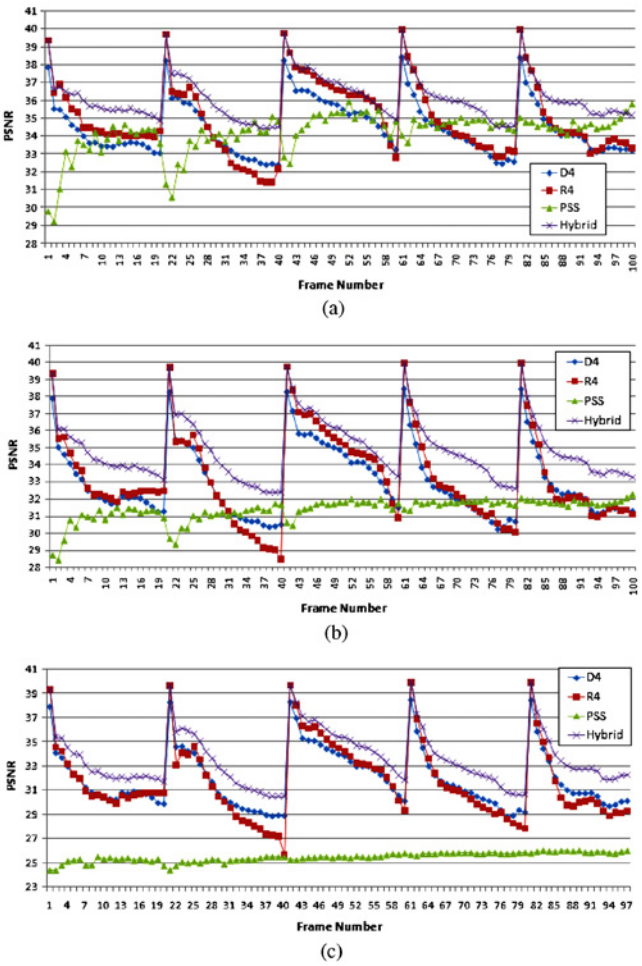


Fig. 19. Frame-by-frame PSNR for missing one, two, and three descriptors. All the MDC methods encoded the stream with the same average bit-rate, 100 kb/s per descriptor. (a) One-descriptor loss. (b) Two-descriptor loss. (c) Three-descriptor loss.

C. Packet-Loss Performance

In this section, the proposed Hybrid method is examined in a packet-loss scenario where various packet-loss rates, ranging from 0% to 15%, are adopted. We compare the Hybrid method with PSS [7] and H.264/AVC, a standard single description coder including basic error concealment as described in [16]. In order to make a fair comparison, both Hybrid and PSS coders encode every video sequence into four descriptors and use one packet for each frame of each descriptor; H.264/AVC encodes every sequence as a single descriptor and uses four packets for each frame. Each packet is lost randomly and independently. Fig. 20 shows the rate-distortion comparison of the three methods for various values of packet-loss rate, P_{loss}. Two QCIF sequences, *Foreman* and *Coastguard*, are used. The results are the averages of 100 independent simulation runs. Fig. 20(a) and (b) presents the results of GOP = 20. It can be seen that H.264/AVC has a better rate-distortion performance than Hybrid for $P_{\text{loss}} < 1\%$, showing that for very low packet-loss rates, the PSNR gain from Hybrid cannot compensate for the loss in coding efficiency. As P_{loss} increases, however, H.264/AVC performance drops quickly but the Hybrid

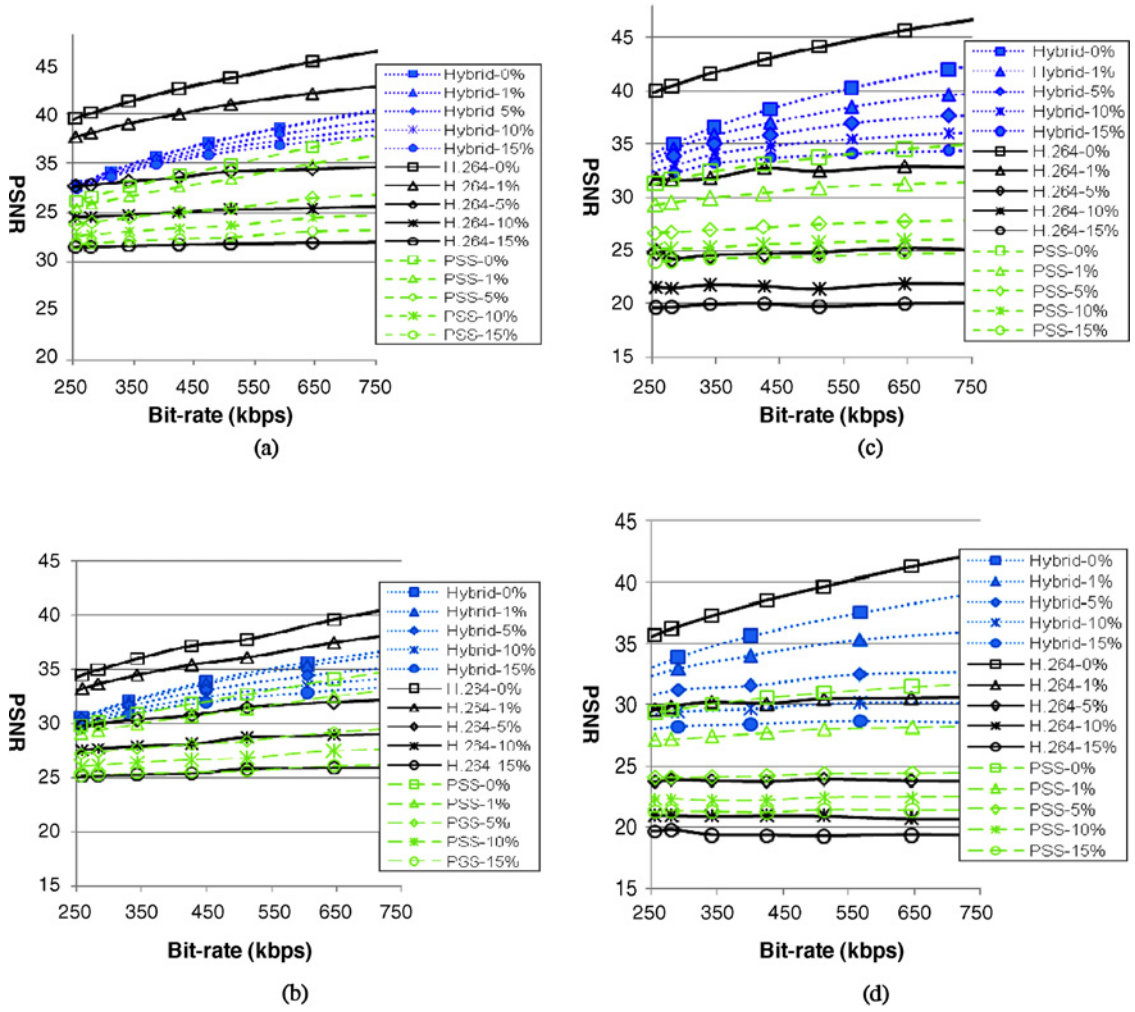


Fig. 20. Performance comparison in packet-loss environment. (a) *Foreman* sequence (GOP = 20). (b) *Coastguard* sequence (GOP = 20). (c) *Foreman* sequence (GOP = 300). (d) *Coastguard* sequence (GOP = 300).

method's performance drops slightly, confirming the error resilience capability of Hybrid. On the other hand, due to high redundancy, PSS outperforms H.264/AVC only when $P_{\text{loss}} > 10\%$. With GOP = 300, Fig. 20(c) and (d) presents the results and it is observed that H.264/AVC suffers a great deal of performance degradation even at low packet-loss rates such as $P_{\text{loss}} = 1\%$; while Hybrid and PSS are not affected that much. This is mainly due to the fact that, with a large GOP size, a single error in a H.264/AVC coded stream may spread out to corrupt the entire frame after a lengthy error-propagation. However, with MDCs such as PSS and Hybrid, the error propagation will be confined to inside the affected descriptor only. Compared with PSS, Hybrid performs better than PSS for all the cases due to its better coding efficiency at the encoder side (note that Hybrid is class-A, while PSS is class-B) and better estimation capability of lost description at the decoder side.

D. Performance of Estimation of Lost Description

This section examines the performance of estimation of lost description in the Hybrid approach. In order to see the effects of combining the spatial and frequency estimation methods (called *Hybrid-SF*), we compare *Hybrid-SF* with *Hybrid-F* and *Hybrid-S*, where Hybrid-S means that, for all the descriptor loss cases, only spatial estimation is applied, and Hybrid-F means only frequency estimation is applied. Other methods used for comparison include *near neighbor replication* (*NNR*), *edge-sensing* (*ES*) [7], and *ES-r*, where NNR is a classical spatial estimation method which replicates the first correctly received pixel in the 8-pixel neighborhood of the current one, starting from the left and proceeding in a clockwise order; ES uses two gradients (ΔH and ΔV) to detect horizontal and vertical edges around the processed pixel, and computes missing pixels while taking the edge orientation into account; ES-r is a variation of ES which, instead of applying estimation of lost description in the pixel domain as in the ES, applies the edge-sensing algorithm on the merged residual data before motion compensation is performed.

The *Foreman* sequence of 200 frames with QP = 28 is used and the results are shown in Table III, where the one-loss and three-loss columns show the average PSNR of four one-descriptor loss cases and four three-descriptor loss cases, respectively. There are six cases for two-descriptor loss; two of them are situations where two descriptors in the same

TABLE III
PSNR OF DIFFERENT ESTIMATION METHODS UNDER VARIOUS DESCRIPTOR-LOSS CASES

	One Loss (4 Cases)	Two Loss (6 Cases)		Three Loss (4 cases)	Average
		Same R	Diff. R		
Hybrid-SF	33.955921	33.519327	32.393669	31.3410015	32.7001137
Hybrid-S	33.519327	33.519327	32.0575	31.3410015	32.4793353
Hybrid-F	33.955921	32.99	32.393669	31.1475	32.5691895
ES-r	33.495	33.495	32.0575	31.325	32.4643373
ES	30.270355	30.27355	32.0575	29.2310465	30.4840117
NNR	26.862746	26.862746	32.0575	26.303388	28.1870784

residual domain (e.g., R0D0 and R0D1) are lost, and four of them are situations where two descriptors in different residual domains (e.g., R0D0 and R1D0) are lost. The average column shows the average PSNR of all 14 cases. As the results show, NNR and ES are the worst performers. Since, among the six methods, NNR and ES are the only two that apply estimation of lost description in the pixel domain, the results indicate that estimation applied in the pixel domain performs worse than those applied in the residual domain. For the cases of one-descriptor loss where only partial coefficients of one residual domain (either R0 or R1) are lost, Hybrid-F performs best since it recovers the missing coefficients by copying the colocated coefficients from the counterpart of the residual domain. Spatial estimation methods such as ES-r and Hybrid-S use the counterpart of residual pixels for recovery, making the performance degradation equal to the cases of two-descriptor loss in the same residual domain (i.e., the Same-R column in the table). For the cases of two-descriptor loss, if two descriptors in the same residual domain are lost, Hybrid-F no longer has advantages over spatial estimation because all coefficients of either R0 or R1 are lost. Spatial estimation methods perform better in these cases, showing that pixel correlation is higher than coefficient correlation. On the other hand, if two descriptors in different residual domains are lost, Hybrid-F performs better since each residual domain has lost partial coefficients only. For the cases of three-descriptor loss, although partial coefficients of one residual domain are obtained, the coefficients in the counterpart of the residual domain are totally lost. In such situations, only pixel correlation can be explored for recovery and therefore, ES-r and Hybrid-S outperform Hybrid-F. It is also observed that in all cases, Hybrid-S performs similarly to ES-r, showing that edge-sensing cannot take effect in the residual domain. From Tables II and III, it is worth noting that since the proposed Hybrid-SF has the best estimation of lost description in each case, it has the best performance overall. The results prove the superiority of combining frequency and spatial estimation methods in the Hybrid MDC method.

E. Adapted to Channel Conditions

The proposed Hybrid method described in Section III-B duplicates only one coefficient (i.e., dc) to each descriptor in the same residual domain; in fact, the number of coefficients to be duplicated can be determined by taking into account channel conditions. In order to evaluate the effects of this parameter on performance, we carried out simulations using different values of N, meaning that the first N coefficients were duplicated (e.g., $N = 5$ meant that dc, ac1, ac2, ac3, and ac4 were duplicated). In Fig. 21, the video quality in terms of PSNR is plotted as a function of bit-rates. We observe that, as N increases, the PSNR improves at the penalty of the bit-rate increasing. For QP = 28, as N increases from 1 to 5, the bit-rate increases about 12 kb/s (from 106 kb/s to 118 kb/s) and the PSNR improves about 0.68 dB, 0.82 dB, and 1.13 dB for the cases of one, two, and three-descriptor loss, respectively. The best performance is obtained in the case of three-descriptor loss. For QP = 35, as N increases from 1 to 5, the bit-rate increases by approximately 2.9 kb/s and the PSNR improves by 0.21 dB, 0.33 dB, and 0.51 dB for the cases of one, two, and three-descriptor loss, respectively. The results again indicate that coefficient duplication is more

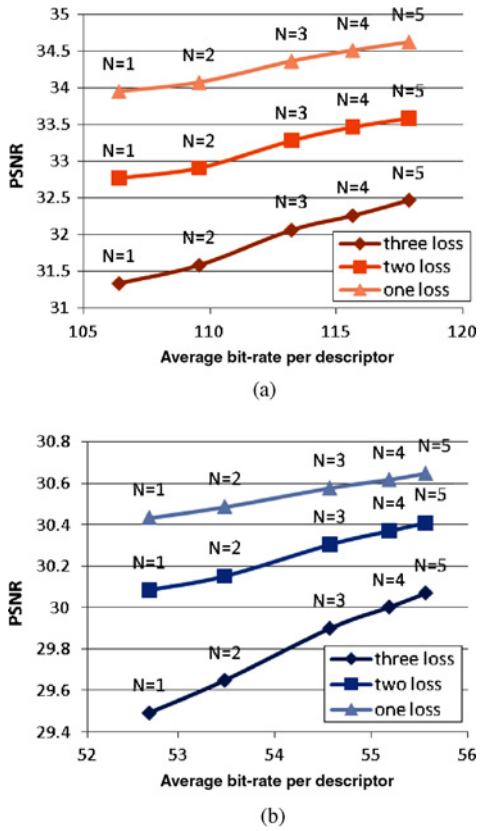


Fig. 21. Performance comparison for various numbers of duplicated coefficients in Hybrid. (a) *Foreman* sequence with QP = 28. (b) *Foreman* sequence with QP = 35.

worth adopting in bad channel conditions. It is worth further noticing that, for QP = 28, the performance/redundancy gains (defined as $\Delta PSNR/\Delta bit\text{-}rate$) for the cases of one, two, and three-descriptor loss are 0.68/12 = 0.057, 0.82/12 = 0.068, and 1.13/12 = 0.094, respectively; while for QP = 35 they are 0.21/2.9 = 0.07, 0.33/2.9 = 0.11, and 0.51/2.9 = 0.176, respectively. The results reveal that coefficient duplication is more beneficial to low bit-rate streams with a larger QP.

V. Conclusion

Two basic and one hybrid MDC method have been proposed. The MDC process in the hybrid encoder is divided into two stages: the first stage splits the residual data into spatial domain, and the second stage splits the ac coefficients into the frequency domain. In the decoder, two types of estimation of lost description, which explore the spatial correlation between residual pixels and the frequency correlation between adjacent blocks, were proposed to improve the reconstruction quality when there are descriptor losses.

The performance evaluation of the proposed MDC methods in both descriptor loss and packet-loss environments has been provided. From the experimental results, it reveals that there is not much performance difference between R4 and D4, no matter for one, two or three-descriptor loss. The difference depended only on the content of the sequences. PSS had the overall worst performance compared with all other methods under the same bit-rates. PSS suffered from a dramatic quality degradation in the cases of two and three-descriptor loss, although it has a relative better performance in one-descriptor loss. Through the design of hybrid encoder, the estimation of lost description in the hybrid decoder is more effective, even when only one or two descriptors are received. The experimental results show that the hybrid method has better performance than all others in most of the cases. We conclude that the proposed hybrid method is adaptive to heterogeneous networks with different number-of-descriptor requirements and also adaptive to dynamic environments with packet loss during transmission.

Acknowledgment

The authors would like to thank Prof. C. W. Chen for his efforts in handling the paper.

References

[1] A. Vetro, J. Xin, and H. Sun, "Error resilience video transcoding for wireless communications," *IEEE Wirel. Commun.*, vol. 12, no. 4, pp. 14–21, Aug. 2005.

[2] V. K. Goyal, "Multiple description coding: Compression meets the network," *IEEE Signal Process. Mag.*, vol. 18, no. 5, pp. 74–93, Sep. 2001.

[3] Y. Wang, A. R. Reibman, and S. Lin, "Multiple description coding for video delivery," *Proc. IEEE*, vol. 93, no. 1, pp. 57–70, Jan. 2005.

[4] V. A. Vaishampayan, "Design of multiple description scalar quantizers," *IEEE Trans. Inf. Theory*, vol. 39, no. 3, pp. 821–834, May 1993.

[5] J. Apostolopoulos, W. Tan, S. J. Wee, and G. W. Wornell, "Modeling path diversity for multiple description video communication," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, vol. 3. May 2002, pp. 2161–2164.

[6] O. Campana and R. Contiero, "An H.264/AVC video coder based on multiple description scalar quantizer," in *Proc. IEEE Asilomar Conf. Signals Syst. Comput. (ACSSC)*, Pacific Grove, CA, 2006, pp. 1049–1053.

[7] R. Bemardini, M. Durigon, R. Rinaldo, L. Celetto, and A. Vitali, "Polyphase spatial subsampling multiple description coding of video streams with H.264," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, vol. 5. Oct. 2004, pp. 3213–3216.

[8] A. Reibman, H. Jafarkhani, Y. Wang, and M. Orchard, "Multiple description video using rate-distortion splitting," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, vol. 1. 2001, pp. 978–981.

[9] K. R. Matty and L. P. Kondi, "Balanced multiple description video coding using optimal partitioning of the DCT coefficients," *IEEE Trans. Circuit. Syst. Video Technol.*, vol. 15, no. 7, pp. 928–934, Jul. 2005.

[10] N. Conci and F. G. B. Natale, "Multiple description video coding using coefficients ordering and interpolation," in *Proc. Signal Process. Image Commun.*, vol. 22, no. 3. Mar. 2007, pp. 252–265.

[11] J. Jia and H. K. Kim, "Polyphase downsampling based multiple description coding applied to H.264 video coding," *IEICE Trans.*, vol. E89-A, no. 6, pp. 1601–1606, Jun. 2006.

[12] J. G. Apostolopoulos, "Error-resilient video compression through the use of multiple states," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, vol. 3. Sep. 2000, pp. 352–355.

[13] S. Gao and H. Gharavi, "Multiple description video coding over multiple path routing networks," in *Proc. Int. Conf. Digital Commun. (ICDT)*, Aug. 2006, p. 42.

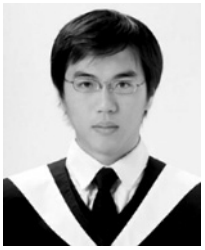
[14] D. Wang, N. Canagarajah, and D. Bull, "Slice group based multiple description video coding using motion vector estimation," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2004, pp. 3237–3240.

[15] H.264/AVC Reference Software: JM13.2 [Online]. Available: http://iphome.hhi.de/suehring/tml

[16] T. Stockhammer, M. M. Hannuksela, and T. Wiegand, "H.264/AVC in wireless environments," *IEEE Trans. Circuit. Syst. Video Technol.*, vol. 13, no. 7, pp. 657–673, Jul. 2003.

[17] T. Tillo, M. Grangetto, and G. Olmo, "Redundant slice optimal allocation for H.264 multiple description coding," *IEEE Trans. Circuit. Syst. Video Technol.*, vol. 18, no. 1, pp. 59–70, Jan. 2008.

[18] N. Zhang, Y. Lu, F. Wu, X. Wu, and B. Yin, "Efficient multiple-description image coding using directional lifting-based transform," *IEEE Trans. Circuit. Syst. Video Technol.*, vol. 18, no. 5, pp. 646–656, May 2008.



Chia-Wei Hsiao received the B.S. and M.S. degrees in computer science from National Chiao-Tung University, Hsinchu, Taiwan, in 2006 and 2008, respectively.

He is with Alpha Image Technology Corporation, Jubei City, Hsinchu, Taiwan as a Researcher in the Multimedia Processing Unit. His research interest includes video codec firmware development.

Wen-Jiin Tsai (M'06–08) received the B.S. and Ph.D. degrees in computer science from National Chiao-Tung University, Hsinchu, Taiwan, in 1992 and 1997, respectively.

She is an Assistant Professor at the Department of Computer Science, National Chiao-Tung University. Before joining National Chiao-Tung University in 2004, she was with Zinwell Corporation, Taipei Hsien, Taiwan, as a Senior Research and Development Manager for six years. Her research interests include video coding, video streaming, error-concealment, and error resilience techniques.