# H.264-Based Multiple Description Coding Using Motion Compensated Temporal Interpolation

Claudio Greco, Marco Cagnazzo, Béatrice Pesquet-Popescu

*TELECOM-ParisTech, TSI department*
*46 rue Barrault, F-75634 Paris Cedex 13*
*Paris, FRANCE*

{greco,cagnazzo,pesquet}@telecom-paristech.fr

***Abstract*—Multiple description coding is a framework adapted to noisy transmission environments. In this work, we use H.264 to create two descriptions of a video sequence, each of them assuring a minimum quality level. If both of them are received, a suitable algorithm is used to produce an improved quality sequence. The key technique is a temporal image interpolation using motion compensation, inspired to the distributed video coding context. The interpolated image blocks are weighted with the received blocks obtained from the other description. The optimal weights are computed at the encoder and efficiently sent to the decoder as side information. The proposed technique shows a remarkable gain for central decoding with respect to similar methods available in the state of the art.**

***Index Terms*—Video coding, multiple description, legacy coder, distributed video coding.**

## I. Introduction

Multiple Description Coding (MDC) is a framework that allows an improved immunity to losses on error prone channels, when no back channel is available or when retransmission delay is not tolerable [1]. Using MDC, robustness is traded off with coding efficiency in terms of compression ratio for a given quality. Given an input signal – image, audio, video, etc. – an MD coder produces a set of independently decodeable, mutually refineable description of equal (or almost equal) rate and importance; each description provides low, yet acceptable, quality; while as any further description is received, the quality of the reconstruction increases, independently on which description it is [2]. The decoding block used when all descriptions are received is referred to as *central decoder*; the decoding block used when any subset of the description is received is referred to as *lateral decoder*.

Several ways to achieve MDC have been explored. Apostolopoulos [3] suggested to split the input sequence in even and odd frames, to be encoded independently. The loss of a frame in one description is recovered estimating a dense motion vector field from the closest frames in the other description, then interpolating the lost frame.

In 2004, Zhang and Stevenson [4] suggested that computing and sending the exact motion vector between frames of two description at the encoder could yield better error recovery performance than motion search, which was, at the time, complex and often inaccurate.

Tillier et al. [5] presented in 2007 a wavelet-based video coder both progressive and MD.

Aside with techniques which aim to design a MD coder *ex-novo*, Shirani et al. [6], [7] pointed out that an MD coder based only on pre-/post-processing and use of legacy coders reduces significantly the development time, hence the development cost. However, this benefit comes at the price of sub-optimal performance with respect to the from-scratch solutions. This idea, originally formulated for still images, has been extended to video coding by Wang et al. [8].

The rest of this paper is organised as follows: in Section II we propose a scheme for two-description coding entirely based on pre- and post-processing; in Section III we present our experimental results in comparison with a recently proposed similar technique; finally, in Section IV we draw conclusion and point some possibilities for future work.

## II. Proposed Coding Scheme for Double Description

The original sequence is split up into even and odd frames; each sub-sequence is then separately encoded with a video coder to produce the two descriptions.

This technique is codec-agnostic, as it only works with decoded frames (before encoding or after decoding); however, we shall assume that H.264, which is the state-of-the-art solution for video coding, is used.

### A. Motion compensated interpolation

Lateral decoding is performed decoding the received description with an H.264 decoder, then reconstructing the missing frames via temporal interpolation; in our scheme, we shall use the DISCOVER technique of temporal interpolation, originally designed for distributed video coding [9], [10].

The interpolation method is summarized in Fig. 1. We call $I_k$ frame to be estimated by using produced by using the temporal adjacent frames $I_{k-1}$ and $I_{k+1}$ which are available from the received description. The reference frames $I_{k-1}$ and $I_{k+1}$ are spatially filtered to smooth out possible noise and higher frequency contributions. Then, a block-matching motion estimation algorithm allows to find a forward motion vector field (MVF) between images $I_{k-1}$ and $I_{k+1}$. A further bidirectional ME is performed in order to find the movement

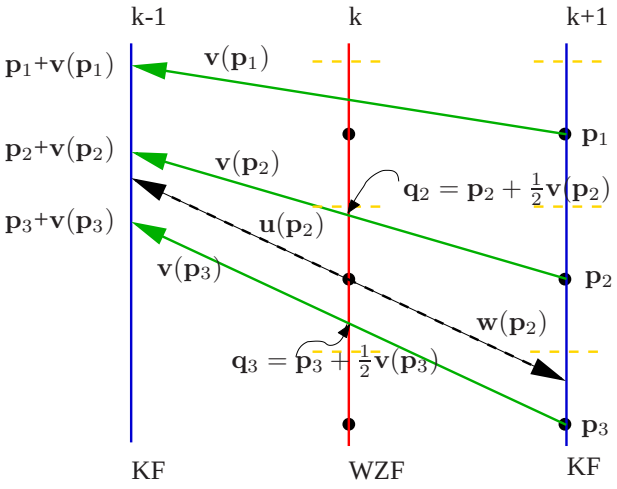


Figure 1. Bidirectional motion estimation in DISCOVER. Green solid arrows: results of forward ME. Black dashed arrows: results of bidirectional ME for the block centred in $\mathbf{p}_2$.

between the current $I_k$ and the references. Let us consider a block of pixels centred in the position $\mathbf{p}_2$ (see Fig. 1). Let $\mathbf{v}$ be the MVF from $I_{k+1}$ to $I_{k-1}$, $\mathbf{u}$ the one from $I_k$ to $I_{k-1}$, and $\mathbf{w}$ the one from $I_k$ to $I_{k+1}$. The motion vector computed by the forward motion estimation is $\mathbf{v}(\mathbf{p}_2)$ and it points to the position $\mathbf{p}_2+\mathbf{v}(\mathbf{p}_2)$ in the frame $I_{k-1}$. The underlying model assumes linear, constant-speed motion, so assume that $\mathbf{u}(\mathbf{p}_2+\frac{1}{2}\mathbf{v}(\mathbf{p}_2))=\frac{1}{2}\mathbf{v}(\mathbf{p}_2)$. However, in order to avoid gaps and overlaps in the motion compensated image, it is needed to estimate $\mathbf{u}(\mathbf{p}_2)$. For this position, the vector closest to the block center is considered. In Fig. 1 this is $\mathbf{v}(\mathbf{p}_3)$, since $\|\mathbf{p}_2-\mathbf{q}_3\|<\|\mathbf{p}_2-\mathbf{q}_2\|$, where $\mathbf{q}_i=\mathbf{p}_i+\frac{1}{2}\mathbf{v}(\mathbf{p}_i)$. In conclusion, in this case the DISCOVER algorithm shall choose:

$$\mathbf{u}(\mathbf{p}_2)=\frac{1}{2}\mathbf{v}(\mathbf{p}_3) \qquad \mathbf{w}(\mathbf{p}_2)=-\frac{1}{2}\mathbf{v}(\mathbf{p}_3) \qquad (1)$$

Finally, there is a further processing of the MVFs: first, it is possible to refine the vector around the position found in Eq. (1). Second, the MVFs are regularized via a weighted median filter. In this way we obtain a couple of MVFs to be used for the motion compensation of $I_{k-1}$ and $I_{k+1}$. The average of the compensated images is the estimation of $I_k$.

### B. Central Decoding

When both decoded descriptions are available, central decoding is performed as a block-wise convex combination of the sub-sequences. For each frame, the relative weight $\alpha_{i,j,k}$ of each block in the received frame with respect to the corresponding block in the interpolated frame is computed at the encoder to minimise the distortion between the block in the original frame and the convex combination; then the sequence of $\alpha$ is sent along with the descriptions as side information. A scheme of the central decoder is shown in Figure 2

The idea of reusing information from the lower fidelity version of a frame in central decoding by means of a convex combination has been originally proposed by Zhu et al. [11]; however, in their work, the lower fidelity frame was a transmitted B frame of lower hierarchical level, whereas we propose to use an interpolated frame generated at the decoder side.

This is a simple technique to obtain an MD codec from H.264 without having access to the codec implementation, which is used as a black-box.

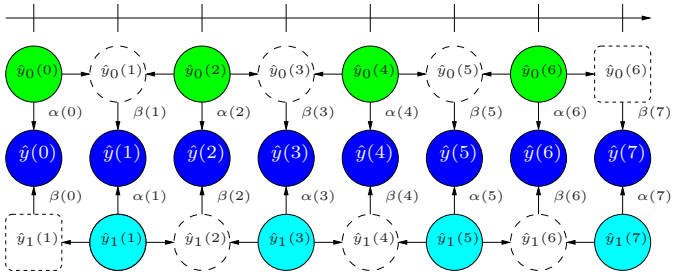


Figure 2. Structure of central decoder. Solid circles represent received frames; dashed circles represent interpolated frames. Horizontal arrows represent interpolation. Vertical arrows represent weighted sum. With $\hat{y}_n(k)$ we denote the $k$-th frame of description $n$. For each $k$, $\beta(k)=1-\alpha(k)$.

### C. Side Information Coding

Even though in theory the relative weight $\alpha$ is a continuous variable, experimental results show that quantizing $\alpha$ on three bits – i.e., eight levels – introduce a negligible on the reconstructed sequence. We shall refer to the quantized version of $\alpha$ with $\bar{\alpha}$.

In order to reduce the bitrate needed to transmit the sequence of weights $\bar{\alpha}$, we adopt a context-based coding. We have found that there is some statistical dependence between the values of $\bar{\alpha}$ and the quantity $E$, defined as the MSE between received blocks and interpolated blocks. This quantity measures the similarity between the two descriptions. When they are very different, usually the received block is a better representation of the original one than the interpolated block. Thus, the mass probability function of $\bar{\alpha}$ is more concentrated around 1, as shown in Fig. 3.

Therefore, the context-based coding has a rate bounded by $H(\bar{\alpha}|E)$, and $H(\bar{\alpha}|E)<H(\bar{\alpha})$ because of the dependency. However, since the number of possible contexts (*i.e.*, of MSE values) is very high, we risk to incur into a *context dilution* problem: having too many contexts makes it difficult or practically impossible to estimate and update the conditional probabilities of symbols during the encoding process.

So we need to perform a context quantization procedure [12]. In other words, instead of using a different entropic coder for each value of $E$, we group the values into clusters defined by a quantization function $Q(E)$. This increase the coding rate, since $H(\bar{\alpha}|Q(E)\geq H(\alpha|E)$. The difference between the two rate bounds (that is the rate penalty) is the mutual information $I(\bar{\alpha},E|Q(E))$. For the sake of simplicity, we use a convex quantizer, that is, the MSE values are grouped into intervals, therefore we only need to choose the thresholds. This is done by minimising the mutual information $I(\bar{\alpha},E|Q(E))$. Given the relatively simple structure of the quantizer, this can be achieved by the means of as simple

| QP | Marginal | Conditional | Gain |
|---|---|---|---|
| 22 | 26.30 | 19.75 | 25.21% |
| 25 | 27.89 | 22.24 | 20.36% |
| 28 | 29.26 | 24.40 | 16.70% |
| 31 | 30.27 | 26.08 | 13.95% |
| 33 | 30.86 | 27.02 | 12.53% |
| 36 | 31.71 | 28.30 | 10.90% |
| 39 | 32.40 | 29.32 | 9.71% |
| 42 | 32.61 | 29.90 | 8.72% |

Table I
BITRATE (IN KBPS) NEEDED TO TRANSMIT THE SIDE INFORMATION WITH STANDARD ENTROPY CODING AND WITH CONDITIONAL ENTROPY CODING GIVEN THE DISTORTION BETWEEN THE TWO DECODED DESCRIPTIONS.

algorithms as the gradient descent, and we do not need more complex iterative techniques as the popular Minimum Conditional Entropy Context Quantization [13] or the improved version called MINIMA [12].

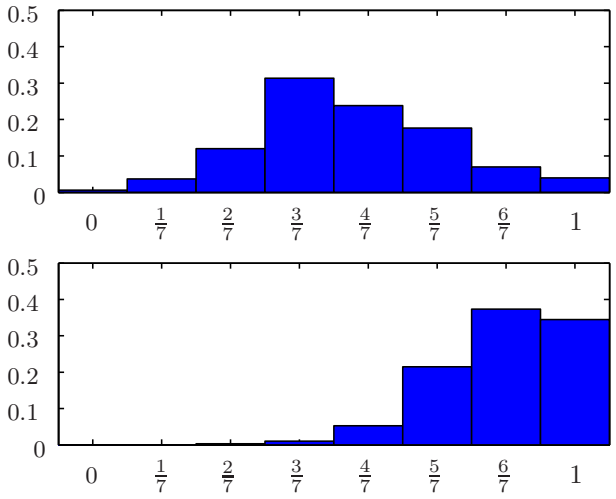


Figure 3. Probability Mass Function of $\bar{\alpha}$ given $Q(E)$ for small (top) and large (bottom) values of $E$.

## III. EXPERIMENTAL RESULTS

We used version 17.0 of the H.264/AVC reference software, JM [14], to encode the sequences in our proposed scheme and in the scheme of Zhu et al. [11], which we took as reference.

The key difference between the two schemes is that the lowest level B-frame in the reference scheme are simply skipped in ours, and replaced at the decoder with an interpolation obtained via DISCOVER temporal interpolation. In both schemes, we used a closed GOP structure with I frames as key frames, which is the most suitable choice for transmission on a lossy channel since it prevents error propagation.

Also, the rate overhead for the side information depends on the entropy of the sequence $\alpha$ in the reference scheme, whereas it depends on the *conditional* entropy of $\alpha$ given the distortion between the two descoded descriptions in ours. The benefit of this strategy of entropy coding can be seen in Table I, where the overheads for two schemes are presented.

A set of eight QPs has been selected (namely 22, 25, 28, 31, 33, 36, 39, and 42) in order to compare the RD performance of the two methods.

The rate-distortion performance comparison for the video sequence "Foreman" (CIF, 30 fps) on the first 128 frames are shown in Figure 4.

According to the metric proposed by Bjontegaard [15], our technique has a gain of 1.80dB in Y-PSNR corresponding to a reduction of 25.84% rate at low bitrates for central decoding. A more extenstive comparison is shown in Table II (do notice that, since sequences are encoded with a fixed QP, the resulting bitrate is higher for sequences with a higher motion content).

This improvement can be explained as the DISCOVER interpolation technique provides a reconstruction of the missing frames better than the very coarse version provided by the lowest level B-frames in the reference method.

Also, even when such a coarse quantization is used, B-frames still need a certain bitrate in order transmit motion vector, mode selection and so on.

It should be expected that the rate-distortion performance of a scheme based on temporal interpolation highly depends on the motion content of the video sequence. In Table II we report the Bjontegaard comparison for several sequences with increasing motion content. It should be noticed that, whereas lateral decoding is severely impaired for sequences with fast movement, central decoding is still efficient, since the low fidelity of lateral sequences is compensated with an appropriate value of $\alpha$.

The rate-distortion curves for sequences "Akiyo" and "Bus" are also shown in figure 5.

A performance comparison with the reference method as a function of the packet loss rate is illustrated in Figure 7-(a). Packet losses are modelled as independent and identically distributed Bernoulli random variables with success probability $p$ equal to the loss rate.

As espected, sequences with higher motion content are more affected by packet loss; however, our technique consistently outperforms the refernce method by 0.5–1.5 dB. In Fig. 7-(b), the two methods are compared over several bitrates for a fixed packet loss rate of 10%, on sequence "Foreman" (CIF, 30 fps). It can be seen how at low bitrates out method outperforms even the lossless reference method.

## IV. CONCLUSIONS

This work presented a simple, yet efficient, framework to implement an MD coding scheme based on pre-/post-processing and legacy codecs.

We showed how temporal interpolation techniques originally developed for distributed video coding may provide an efficient tool for lateral decoding in temporal splitting based MD methods. Furthermore, we showed how this method is adapted to side information aided central decoding, and how the correlation between the side information and the distortion between the descriptions can be exploited in order to reduce the bitrate.

| Sequence | Bitrate range $[kbps]$ | Y-PSNR Gain (Central) | Bitrate Variation (Central) | Y-PSNR Gain (Lateral) | Bitrate Variation (Lateral) |
|---|---|---|---|---|---|
| akiyo | $39 \sim 67$ | $+4.85$dB | $-42.15\%$ | $+3.84$dB | $-51.48\%$ |
| akiyo | $56 \sim 100$ | $+2.95$dB | $-35.51\%$ | $+2.06$dB | $-39.34\%$ |
| akiyo | $77 \sim 194$ | $+1.65$dB | $-29.54\%$ | $+1.36$dB | $-32.66\%$ |
| hall | $58 \sim 105$ | $+3.21$dB | $-34.42\%$ | $+2.53$dB | $-47.93\%$ |
| hall | $83 \sim 178$ | $+1.42$dB | $-26.98\%$ | $+1.28$dB | $-35.52\%$ |
| hall | $127 \sim 458$ | $+0.60$dB | $-20.91\%$ | $+0.69$dB | $-32.98\%$ |
| foreman | $95 \sim 190$ | $+1.80$dB | $-25.84\%$ | $+1.56$dB | $-32.85\%$ |
| foreman | $148 \sim 317$ | $+1.35$dB | $-24.88\%$ | $+1.22$dB | $-31.80\%$ |
| foreman | $231 \sim 657$ | $+1.08$dB | $-23.36\%$ | $+1.01$dB | $-32.88\%$ |
| city | $82 \sim 190$ | $+1.53$dB | $-24.04\%$ | $+1.33$dB | $-32.13\%$ |
| city | $141 \sim 336$ | $+1.01$dB | $-17.38\%$ | $+0.73$dB | $-17.98\%$ |
| city | $238 \sim 726$ | $+0.67$dB | $-12.74\%$ | $+0.16$dB | $-05.19\%$ |
| flower | $154 \sim 511$ | $+1.07$dB | $-20.76\%$ | $+0.89$dB | $-21.96\%$ |
| flower | $342 \sim 1072$ | $+0.77$dB | $-14.87\%$ | $+0.67$dB | $-17.08\%$ |
| flower | $691 \sim 2322$ | $+0.72$dB | $-11.92\%$ | $+0.65$dB | $-15.91\%$ |
| mobile | $156 \sim 422$ | $+1.05$dB | $-20.79\%$ | $+0.81$dB | $-21.44\%$ |
| mobile | $286 \sim 931$ | $+0.59$dB | $-14.66\%$ | $+0.15$dB | $-05.68\%$ |
| mobile | $572 \sim 2401$ | $+0.62$dB | $-13.76\%$ | $-0.08$dB | $+2.73\%$ |
| stefan | $166 \sim 416$ | $+1.08$dB | $-15.98\%$ | $+0.65$dB | $-13.57\%$ |
| stefan | $305 \sim 786$ | $+0.53$dB | $-10.02\%$ | $+0.00$dB | $-01.88\%$ |
| stefan | $544 \sim 1791$ | $+0.36$dB | $-07.12\%$ | $-0.36$dB | $+11.46\%$ |
| coastguard | $93 \sim 343$ | $+0.61$dB | $-19.08\%$ | $+0.48$dB | $-21.00\%$ |
| coastguard | $208 \sim 830$ | $+0.40$dB | $-11.96\%$ | $+0.24$dB | $-09.44\%$ |
| coastguard | $494 \sim 2058$ | $+0.43$dB | $-09.72\%$ | $+0.30$dB | $-09.60\%$ |
| bus | $172 \sim 473$ | $+0.83$dB | $-16.25\%$ | $-0.31$dB | $+07.11\%$ |
| bus | $335 \sim 890$ | $+0.73$dB | $-13.41\%$ | $-0.83$dB | $+23.70\%$ |
| bus | $608 \sim 1861$ | $+0.66$dB | $-11.54\%$ | $-1.27$dB | $+37.98\%$ |
| football | $191 \sim 574$ | $+0.59$dB | $-13.43\%$ | $-1.05$dB | $+38.92\%$ |
| football | $399 \sim 1062$ | $+0.41$dB | $-08.42\%$ | $-1.44$dB | $+54.56\%$ |
| football | $737 \sim 2083$ | $+0.39$dB | $-06.63\%$ | $-1.62$dB | $+54.28\%$ |

Table II

BJONTEGAARD COMPARISON BETWEEN PROPOSED AND REFERENCE TECHNIQUE FOR VARIOUS SEQUENCES WITH INCREASING MOTION CONTENT.

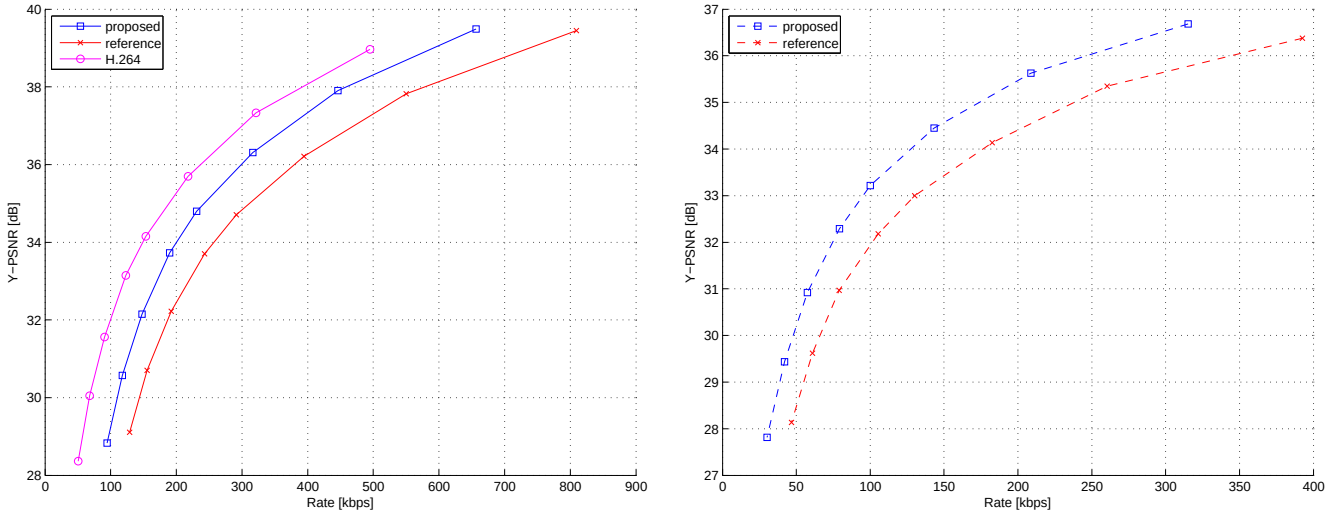


Figure 4. Comparison of the proposed method against the Hierarchical B-Picture Coding in [11] for central (left) and lateral (right) decoder (sequence "Foreman").

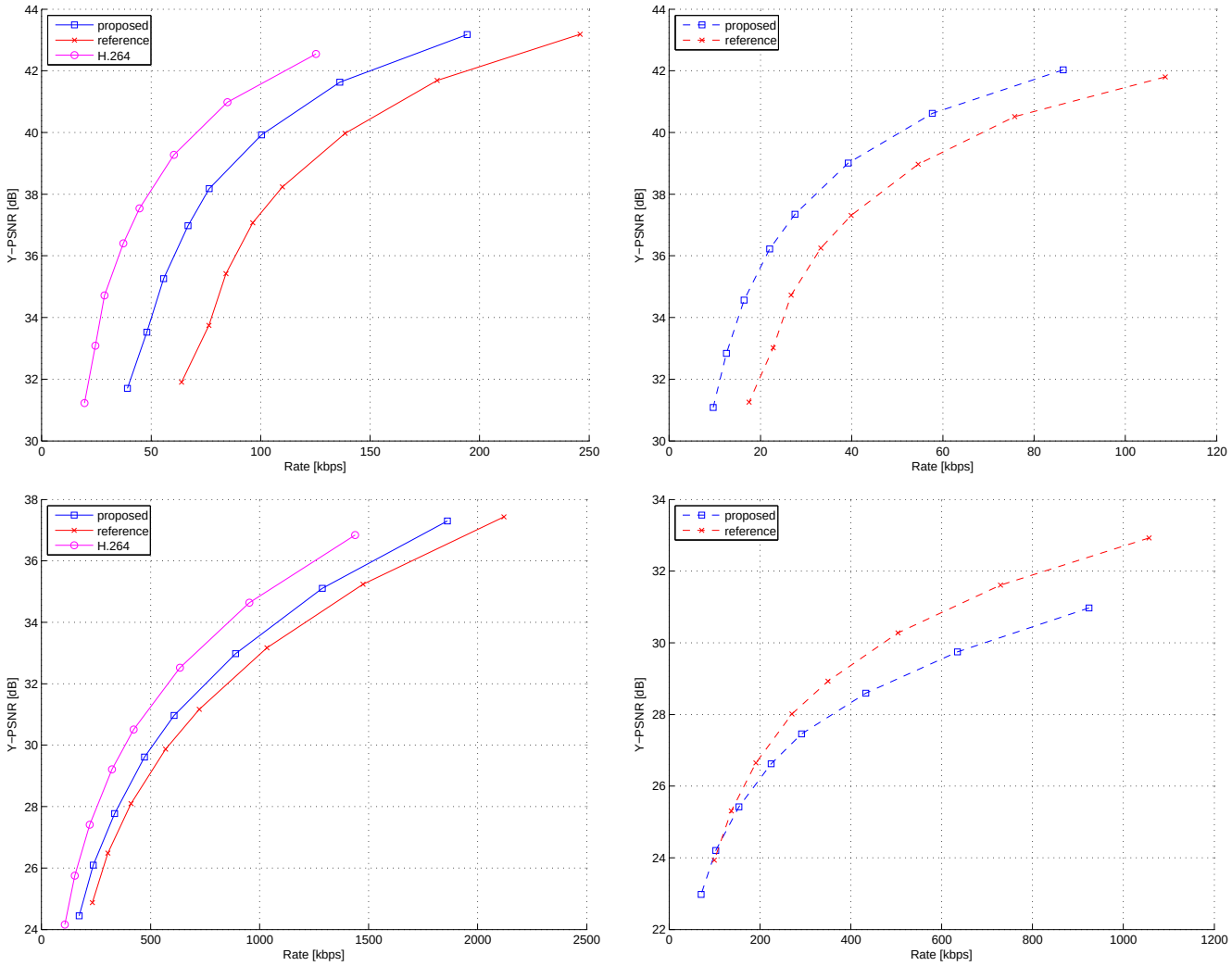


Figure 5. Comparison of the proposed method against the Hierarchical B-Picture Coding in [11] for sequences "Akiyo" (top) and "Bus" (bottom); central (left) and lateral (right) decoder.


Figure 6. Visual comparison of frame 86 of sequence "Foreman" (CIF sequence, 30 fps, 256 kbps) in reference (left, Y-PSNR=31.67dB) and proposed (right, Y-PSNR=33.55dB) method.

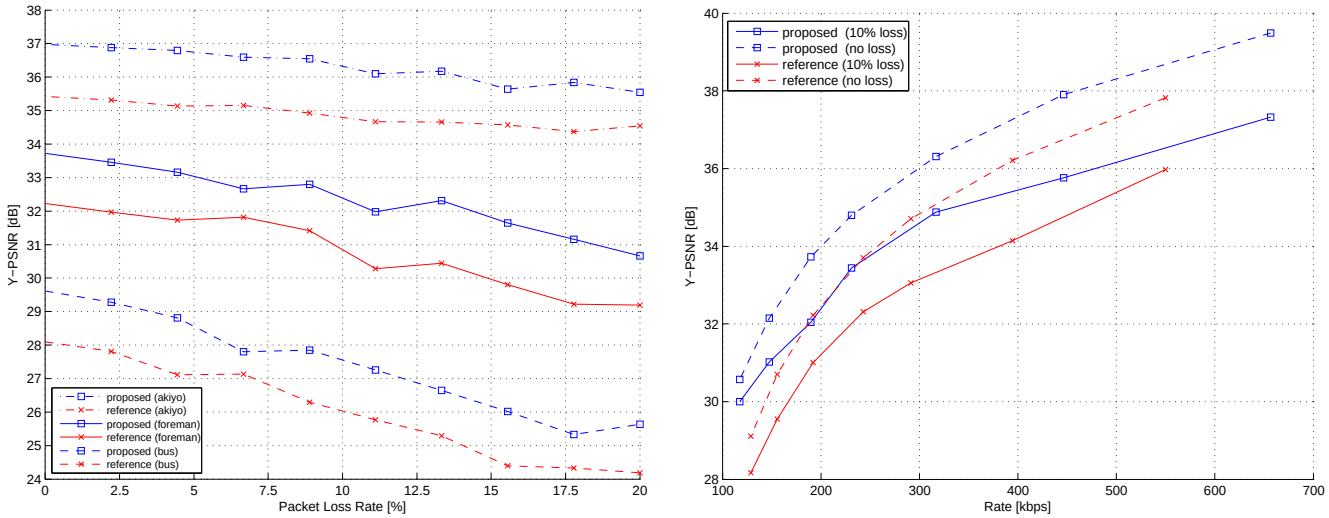


Figure 7. (a) Performance versus packet loss rate comparision for fixed bitrate (200 kbps). (b) Performance versus bitrate comparison for fixed packet loss rate (10%).

Future research on this subject includes the extension of the technique to more than two descriptions. Also, higher order motion interpolation, such as the one proposed by Petrazzuoli et al. [16], could be used instead of DISCOVER to improve the performance of both side and central decoding.

## REFERENCES


[1] A. El Gamal and T. Cover, "Achievable rates for multiple descriptions," *IEEE T Inform Theory*, vol. 28, no. 6, pp. 851–857, November 1982.

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

[3] J. Apostolopoulos, "Reliable video communication over lossy packet networks using multiple state encoding and path diversity," in *VCIP '01: Visual Communications and Image Processing*, 2001, pp. 392–409.

[4] G. Zhang and R. Stevenson, "Efficient error recovery for multiple description video coding," in *ICIP '04: International Conference on Image Processing*. IEEE, 2004.

[5] C. Tillier, T. Petrişor, and B. Pesquet-Popescu, "A motion-compensated overcomplete temporal decomposition for multiple description scalable video coding," *EURASIP J Im Vid Proc*, 2007.

[6] S. Shirani, M. Gallant, and F. Kossentini, "Multiple description image coding using pre- and post-processing," in *ITCC '01: International Conference on Information Technology: Coding and Computing*. IEEE, 2001.

[7] E. Kozica, D. Zachariah, and W. Kleijn, "Interlacing intraframes in multiple-description video coding," in *ICIP '07: International Conference on Image Processing*. IEEE, 2007.

[8] D. Wang, N. Canagarajah, D. Redmill, and D. Bull, "Multiple description video coding based on zero padding," in *ISCAS '04: International Symposium on Circuits and Systems*. IEEE, 2004.

[9] C. Guillemot, F. Pereira, L. Torres, T. Ebrahimi, R. Leonardi, and J. Ostermann, "Distributed monoview and multiview video coding," *IEEE Signal Proc Mag*, vol. 24, pp. 67–76, September 2007.

[10] X. Artigas, J. Ascenso, M. Dalai, S. Klomp, D. Kubasov, and M. Ouaret, "The DISCOVER codec: Architecture, techniques and evaluation," in *Picture Coding Symposium*, 2007.

[11] C. Zhu and M. Liu, "Multiple description video coding based on hierarchical B pictures," *IEEE T Circ Syst Vid*, vol. 19, no. 4, pp. 511–521, April 2009.

[12] M. Cagnazzo, M. Antonini, and M. Barlaud, "Mutual information-based context quantization," *Elsevier Signal Processing: Image Communication*, vol. 25, no. 1, pp. 64–74, January 2010.

[13] X. Wu, P. A. Chou, and X. Xue, "Minimum conditional entropy context quantization," Sorrento, Italy, June 2000.

[14] "H.264/avc JM reference software," Website. [Online]. Available: http://iphome.hhi.de/suehring/tml/

[15] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," in *VCEG Meeting*, Austin, USA, Apr. 2001.

[16] G. Petrazzuoli, M. Cagnazzo, and B. Pesquet-Popescu, "High order motion interpolation for side information improvement in DVC," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, Dallas, TX, 2010.