

# A DISTRIBUTED-SOURCE-CODING BASED ROBUST SPATIO-TEMPORAL SCALABLE VIDEO CODEC

*Marco Tagliasacchi*

Politecnico di Milano, Italy

*Abhik Majumdar, Kannan Ramchandran*

University of California, Berkeley

## ABSTRACT

A video coding paradigm called PRISM (Power-efficient, Robust, hIgh compression, Syndrome-based Multimedia coding) built on distributed source coding principles has been recently proposed in [1]. In this paper, we study a scalable version of PRISM which addresses both spatial and temporal scalability. The proposed codec inherits the main attributes of the PRISM architecture, namely increased robustness to channel losses and more flexible sharing of computational complexity between encoder and decoder, while adding scalability as an additional feature. In this paper, we show the efficacy of enhancing a standards-compliant base layer codec (H.264/AVC) with a PRISM refinement bit stream, resulting in an overall robust spatio-temporal scalable video codec that is backward-compatible with the existing standards-based baseline codec.

## 1. INTRODUCTION

Robust scalable video coding has become an important problem in light of the recent proliferation of multimedia applications over wireless networks. The wireless medium requires robustness to channel losses. At the same time, scalability is important in many applications like multicast, surveillance, and browsing. Spatial and temporal scalability, together with superior resilience to packet losses, can be useful for example in broadcasting to a set of heterogeneous mobile receivers having varying computational and display capabilities and/or channel capacities.

While there has been significant interest in scalable video coding in the research community, there is relatively less work on achieving scalability while being additionally robust to channel losses<sup>1</sup>. It is this combination that motivates this work. Classical motion-compensated prediction based video codecs (of the MPEGx/H.26x variety) are inherently fragile in the face of channel losses, since the loss of the predictor leads to errors in subsequent frames. The delay and latency constraints of the video application may further limit the use of Automatic Repeat Request (ARQ) and

Forward Error Correction (FEC) schemes to recover from loss. ARQ schemes also require a feedback channel and are ill-suited to multicast/broadcast scenarios. FEC-based schemes can mitigate the probability of error but can never guarantee error-free operation: when errors do occur, they propagate annoyingly through the video until the next intra-frame is received. Delay and latency constraints may also limit the blocklength (and therefore coding strength) of the FECs, limiting their usefulness. Even when FECs are used, therefore, there is need to mitigate the effects of error propagation (the so-called drift problem).

In response to the challenge of overcoming the fragility of predictive video coding based frameworks, a number of solutions have been proposed recently [1, 5, 6, 10]. While the codecs proposed in [1, 5, 6, 10] differ quite considerably, they are all based on variants of the principles of distributed source coding from multi-user information theory [17]. In this work, we will use the PRISM [1] video coder from the above class, as the basis for building our proposed scalable video coding solution. A key attribute of PRISM [1] is to effect intra-frame coding, while approaching the coding efficiency of motion-compensated inter-frame coding. Rate savings are achieved in PRISM over pure intra-frame-based coding (such as M-JPEG) through carefully exploiting the correlated side-information (in the form of previous temporal video frames) present at the decoder. PRISM differs from classical inter-frame video codecs in two ways in that it allows for flexible sharing of the motion-search complexity between encoder and decoder. Since PRISM allows for the concept of a motion search at the decoder, multiple predictors can be used to decode a video block, unlike as in MPEG, where decoding is tied to a specific predictor. As PRISM is not attached to any specific predictor, it enjoys superior error-resilience properties, and has an intra-frame-like character with regard to error propagation. The PRISM codec has been found to significantly outperform standard video coders, such as H.263+ for transmission over packet loss channels.

Building on the PRISM framework, we propose a coding scheme to provide spatial and temporal scalability based on the principles of distributed video coding. This scalable video coder is designed specifically to provide good per-

---

This work was funded in part by NSF grant CCR-021-9722.

<sup>1</sup>Exceptions are the recent works of [11, 12].

formance in the face of channel losses. While the PRISM framework allows for a flexible distribution of the motion search task between encoder and decoder, in this work we will focus on the case when the encoder does very little motion estimation and most of the motion compensation task is performed at the decoder. This is of particular relevance to emerging “uplink” multimedia applications (such as users streaming video from their cellphones).

Recently, video codecs based on distributed source coding and with scalability properties have been proposed in [11, 12]. However, both these codecs target SNR scalability. On the other hand, in this paper, we target spatio-temporal scalability. Further, our scalable codec relies heavily on the concept of motion search at the decoder, unlike [11, 12].

## 2. BACKGROUND ON PRISM

The PRISM video coder is based on a modification to the source coding with side-information paradigm, where there is inherent uncertainty in the state of nature characterizing the side information (a sort of “universal” Wyner-Ziv framework, see [9] for details). The Wyner-Ziv Theorem [7] deals with the problem of source coding with side-information. The encoder needs to compress a source  $X$  when the decoder has access to a source  $Y$ .  $X$  and  $Y$  are correlated sources and  $Y$  is available only at the decoder. From information theory we know that for the MSE distortion measure and  $X = Y + N$  where  $N$  has a Gaussian distribution, the rate - distortion performance for coding  $X$  is the same whether or not the encoder has access to  $Y$  [8].

For the problem of source coding with side information, the encoder needs to encode the source within a distortion constraint, while the decoder needs to be able to decode the encoded codeword subject to the correlation noise (between the source and the side-information). While, the results of Wyner and Ziv [7] are non-constructive and asymptotic in nature, a number of constructive methods to solve this problem have since been proposed (such as in [14, 15, 16]) wherein the source codebook is partitioned into cosets of a channel code.

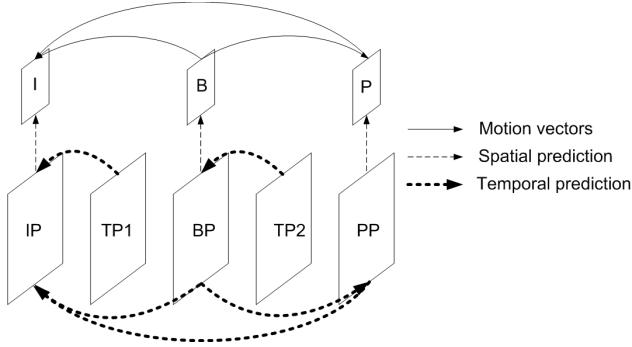
For the PRISM video coder [1], the video frame to be encoded is first divided into non-overlapping spatial blocks of size  $8 \times 8$ . The source  $\mathbf{X}$  is the current block to be encoded. The side-information  $\mathbf{Y}$  is the best (motion-compensated) predictor for  $\mathbf{X}$  in the previous frame and let  $\mathbf{X} = \mathbf{Y} + \mathbf{N}$ . We first encode  $\mathbf{X}$  in the intra-coding mode to come up with the quantized codeword for  $\mathbf{X}$ . Now, we do the syndrome encoding, i.e., we find a channel code that is matched to the “correlation noise”  $\mathbf{N}$ , and use that to partition the source codebook into cosets of that channel code. The encoder transmits the syndrome (indicating the coset for  $\mathbf{X}$ ) and a CRC check of the quantized sequence. In contrast to MPEG, H.26x, etc., it is the decoder’s task to do motion

search, as it searches over the space of candidate predictors one-by-one to decode a sequence from the set labeled by the syndrome. When the decoded sequence matches the CRC check, decoding is declared to be successful. For further details please refer to [2].

**Robustness Characteristics of PRISM** : The key aspect here is that PRISM does not use the exact realization of frame  $(N - 1)$  while encoding blocks in frame  $N$ , but only the correlation statistics. Note that if the decoder does not have frame  $(N - 1)$  (or a part of it) due to channel loss, it might still have blocks from frame  $(N - 2)$ . If the correlation noise between any of these blocks and the current block is within the noise margin for which the syndrome code was designed the current block can be decoded. Informally speaking, PRISM sends specific bit-planes (or partial bit-planes) of the current block, unlike predictive coders which send information about the *difference* between the block and its predictor. Consequently, in the PRISM framework, every time a block can be decoded, it has the same effect as an “intra-refresh” (irrespective of any errors that may have occurred in the past). On the other hand, for predictive coders, once an error occurs, the only way to recover is through an intra-refresh.

## 3. CODEC DETAILS: GOP STRUCTURE

In order to achieve scalability within the PRISM framework, spatio-temporal enhancement layers are encoded on top of a base layer following the guidelines that have recently emerged within the MPEG Ad-Hoc group on Scalable Video Coding (MPEG-SVC). Figure 1 depicts the GOP structure adopted in our codec, which has been chosen to conform with the scalability layers defined in [3] by the MPEG-SVC community. No assumptions are made about the encoding strategy at the base layer. This allows any existing standard, e.g. the state-of-the-art H.264/AVC codec, to be used at the base layer. In this work we have used H.264/AVC to encode the base layer with an IBPBP structure so that the first temporal scalability layer is supported. For example, if the full spatio-temporal resolution sequence is CIF@30fps, then by decoding the base layer only, we obtain a sequence at QCIF@15fps or QCIF@7.5fps (by skipping the B frames). The enhancement layers are encoded on top of the base layer with the proposed distributed source coding based codec. The frames labeled IP, BP and PP form the spatial enhancement layer (achieving CIF@15fps) and these frames can leverage the base layer as a spatial predictor. Subsequently the temporal enhancement layer is added (frames TP1, TP2) in order to obtain the full resolution version of the sequence - CIF@30fps. In both cases, the motion information available in the base layer is exploited to build the temporal predictor as will be detailed in the following sections.



**Fig. 1.** GOP structure. First, the base layer (I, B and P frames) is encoded. Then, a spatial enhancement layer (IP, BP and PP frames) is built on top of the base layer. Lastly, a temporal enhancement layer is added (TP1 and TP2). Solid arrows represent the motion vectors estimated at the base layer, which are also used as coarse motion information at the enhancement layer. Dashed/dotted arrows point to the frame used as reference to build the spatial/temporal predictor.

#### 4. SPATIAL ENHANCEMENT LAYER ENCODING

In the proposed codec, each block<sup>2</sup> is encoded independently with the previously decoded blocks at the decoder serving as the side-information. Since the enhancement layer encoder is not allowed to perform any motion search, the correlation noise between the current block and the unknown best predictor, that will be revealed only during decoding, needs to be computed in a computationally efficient manner. To this end, in the original (non-scalable) version of PRISM [1], each block is classified according to the mean square error computed using as a predictor the block in the reference frame at the same spatial location, i.e. assuming zero motion. An offline training scheme provides an estimate of the correlation noise for each DCT coefficient based on the measured MSE. Unfortunately this method is likely to fail when there is significant, yet simple, motion such as camera panning. The proposed solution takes advantage of the existence of the base layer in two different aspects: as a spatial predictor for those blocks that cannot be properly temporally predicted, e.g. because of occlusions, as well as using the motion vectors available at the coarser resolution to provide a better estimate of the correlation noise. The encoding process for the three types of frames are as follows:

- *Frame IP*: A spatial predictor is computed by interpolating the quantized I frame of the base layer. The prediction residual is quantized and entropy coded as in H.263+.
- *Frame PP*: Spatial, temporal and spatiotemporal predictors are built using only the coarse motion vec-

tors of the base layer. Then, the best predictor is chosen according to a MSE metric (other metrics may also be used) and the correlation noise is estimated based on the statistics collected offline. The block is then quantized and encoded as described in [4], sending only the least significant bits of the DCT coefficients as the most significant ones will be recovered using the side information.

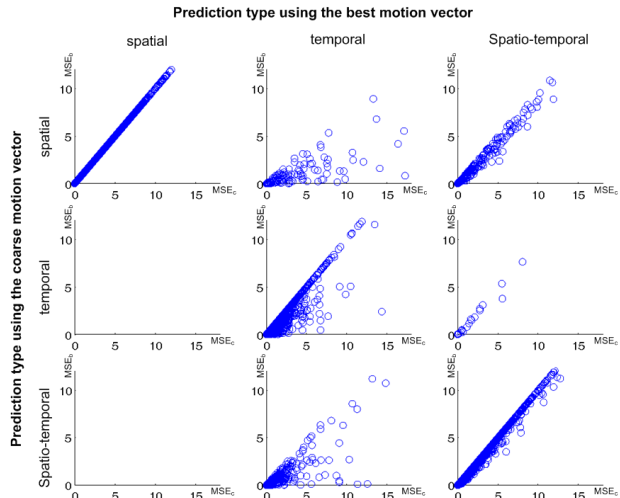
- *Frame BP*: The encoding algorithm is similar to that of the PP frame, except for the fact that the temporal predictor can use the forward and/or backward motion vectors (bi-directional prediction). The prediction mode as well as the motion vectors are the same used in the base layer.

At the decoder, the algorithm tests different predictors until it finds one that is good enough to correctly decode the block. As in [1], a CRC of sufficient strength is used to determine when the block is correctly decoded. A CRC hash is computed from the quantized DCT coefficients and transmitted to the decoder. For each candidate predictor the decoder decodes the block. If the CRC of the decoded block matches the one computed at the encoder side, a decoding success is declared. The decoder is allowed to use any predictor - spatial, temporal and spatiotemporal - for the purpose of decoding the encoded block.

As mentioned above, the correlation between the block to be encoded and the (unknown) best predictor (which will be found at the decoder after a full motion search) needs to be estimated. This is the task of the classifier. At the encoder, only the motion information available at the base layer (termed the “coarse” motion vector) is used to provide an estimate of the correlation. Three different prediction types are allowed - spatial, temporal and spatio-temporal - and the best among these choices is computed based on the “coarse” motion vector. The classifier works offline on training sequences. Using the data collected offline, the classifier estimates the correlation statistics for each type of predictor that can be selected at the encoder (spatial, temporal and spatio-temporal). See Figure 2 for more details.

Although only two levels are considered in the current implementation, the proposed scheme supports any number of levels of spatial scalability. In fact, the same concepts can be extended to a multi-layer architecture, where each layer can use the upper layer as a spatial predictor. Furthermore the ratio between the resolutions of two succeeding layers is not constrained to be 2:1. All that is needed is an interpolating algorithm that is able to build the spatial predictor of the appropriate size starting from the base layer.

<sup>2</sup>8x8 blocks were used in this implementation



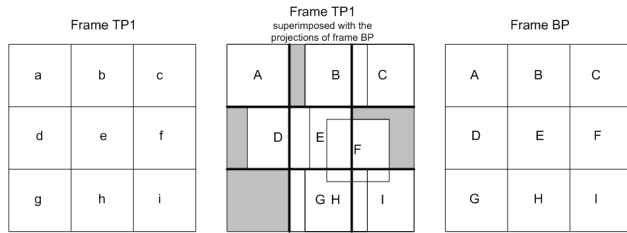
**Fig. 2.** Working offline, the classifier computes a mapping between the residue computed using the “coarse” predictor and the residue computed using the best predictor obtained from a full motion search. Each block is represented by a circle in one of the nine scatter plots, according to the prediction type computed using the coarse motion vector available at the encoder (determining the row) and the best motion vector (determining the column). In each scatter plot, the x-axis is the MSE computed using the coarse motion vector ( $MSE_c$ ), while the y-axis the MSE computed with the best motion vector ( $MSE_b$ ).

## 5. TEMPORAL ENHANCEMENT LAYER ENCODING

The encoding of the temporal enhancement layer is more involved since we can rely only partially on the information available at the base layer. Specifically, we have neither a spatial predictor available nor a motion field that covers completely the frame to be encoded. For these reasons we allow only temporal prediction. The motion field is inferred by that available at the base layer. In our current implementation, the estimation of the coarse motion field for TP1 frames (see Figure 1) proceeds as follows. First, the motion field of frame BP is extracted from the base layer by simply scaling the motion vectors in order to match the spatial resolution of the enhancement layer<sup>3</sup>. Then, the motion field of frame TP1 is estimated by “projecting” the blocks along the motion trajectories from BP to IP (or PP). Figure 3 gives a pictorial representation of the different scenarios that can occur.

The estimation of the motion field of frame TP2 follows the same algorithm as that for frame TP1. In this case we can leverage either the backward motion field from PP to IP (or PP) or the forward motion field from BP to PP. We note that separate statistics of the correlation noise are col-

<sup>3</sup>More sophisticated methods may also be used for this operation



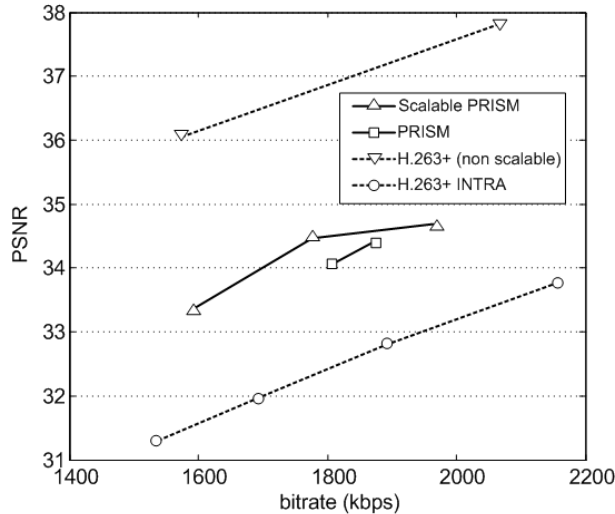
**Fig. 3.** Estimation of motion field of frame TP1 from motion field of frame BP. Block  $e$  is covered by the projections of blocks  $E$ ,  $D$  and  $F$ . The block with the maximum overlap, i.e.  $E$ , is selected and so  $MV_e = MV_E/2$ . Similarly,  $MV_a = MV_A/2$ ,  $MV_b = MV_B/2$  and  $MV_c = MV_C/2$ .

lected for each type of frame. This is due to the fact that the distance between the current frame and its temporal reference is different for each type of frame. Hence the accuracy of the estimated coarse motion field varies with frame type (typically the motion fields estimated for the frames of type BP and PP are more precise than for the frames of type TP1 and TP2).

## 6. EXPERIMENTAL RESULTS

We carried out several experiments in order to assess the performance of the proposed solution. In our simulations we encoded the base layer using H.264/AVC with a IBPBP structure and a GOP of size 8. In the current implementation we force H.264/AVC to work with fixed block sizes of  $8 \times 8$ , as this way it is easier to compute the coarse motion field estimate for the enhancement layer. The spatial and temporal enhancement layers are encoded as described in Sections 4 and 5 respectively. Figure 5 shows the PSNR as a function of the bitrate for the Football sequence. We can observe that the scalable version outperforms the previous implementation of PRISM when working at full spatio-temporal resolution. This can be attributed to the base layer being encoded using the highly efficient H.264/AVC coder and the availability of the motion information in the base layer allowing for better classification of the blocks than using a simple zero-motion predictor. This is especially true in case of sequences characterized by significant camera panning. In the event of a decoding failure, conventional error concealment techniques can be used. The current block can be substituted with its spatial predictor (for BP and PP frames) or its temporal predictor, using the coarse motion vector extracted from the base layer.

Apart from pure compression performance, we also tested the robustness characteristics of three schemes: the proposed scalable video codec, H.264/AVC, and H.264/AVC protected with Forward Error Correcting (FEC) codes using a wireless channel simulator obtained from Qualcomm, Inc. For the H.264/AVC codec protected with FECs, Reed-

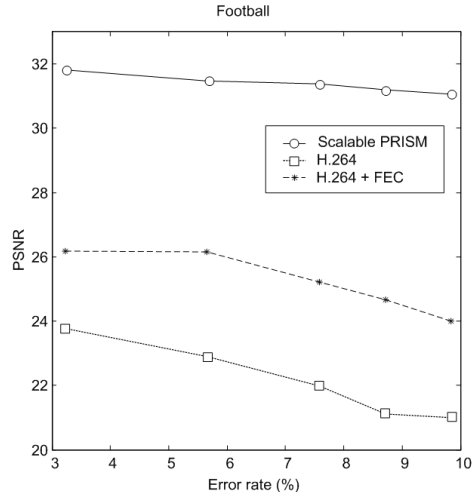


**Fig. 4.** PSNR results for the Football sequence(15fps) at CIF resolution.

Solomon codes were used with 20% of the total rate used for parity bits. This simulator adds packet errors to multimedia data streams transmitted over wireless networks conforming to the CDMA2000 1X standard [13]. As can be seen from Figures 5 and 6, the scalable PSNR implementation clearly out-performs H.264/AVC and even H.264/AVC protected with parity bits by a wide margin (about 8 dB and 6 dB respectively). Figure 7 shows the evolution of PSNR over time for the football video sequence for a loss rate of 7.5%. Figure 8 shows the reconstruction of a particular frame (the middle frame of the GOP) of the Stefan sequence by the proposed scalable PRISM coder and H.264/AVC. As can be seen from Figure 8 the visual quality provided by the scalable PRISM coder is clearly superior to that provided by H.264/AVC. In the channel simulations we allow base layer and enhancement layer packets of the scalable PRISM bit-stream to be dropped. As can be seen from Figures 5, 6 and 8, the scalable PRISM coder is able to provide good quality reconstruction even when parts of the base layer is lost. This is in marked contrast to standard (prediction-based) scalable video coders where loss of the base layer often severely affects the video quality. Currently, we are in the process of designing the codec to work efficiently at lower encoding rates and running extensive tests over different types of channels to further validate our approach.

## 7. CONCLUSIONS

In this paper we propose a scalable coding scheme based on the PRISM framework. We achieve both spatial and temporal scalability while improving the coding efficiency at full resolution. Channel simulations indicate that the pro-

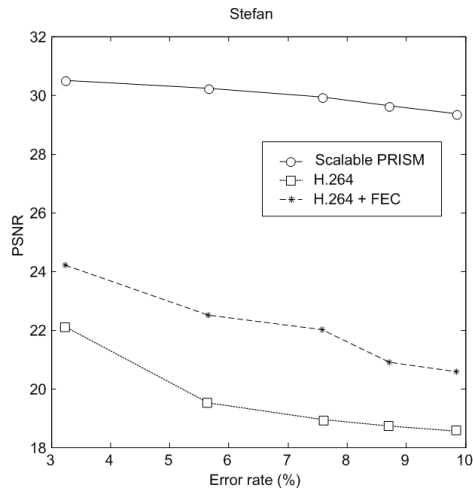


**Fig. 5.** Performance comparison of proposed scalable codec, H.264/AVC, and H.264/AVC protected with FECs (Reed-Solomon codes used, 20% of the total rate used for parity bits) for the Football sequence (CIF, 15fps, 1800 kbps).

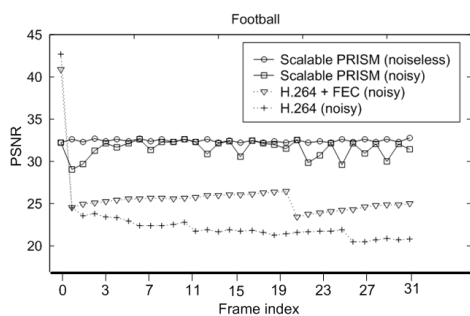
posed scalable PRISM codec is far more resilient to packet losses than conventional predictive codecs and even outperforms, by a significant margin, predictive coded bit-streams protected with Forward Error Correcting codes under reasonable latency constraints.

## 8. REFERENCES

- [1] R. Puri and K. Ramchandran, "PRISM: A New Robust Video Coding Architecture based on Distributed Compression Principles," in *Allerton Conference on Communication, Control and Computing*, 2002.
- [2] R. Puri and K. Ramchandran, "A video coding architecture based on distributed compression principles," in *ERL Technical Report, Memorandum No. UCB/ERL M03/6*, Mar 2003.
- [3] MPEG Ad Hoc Group on Scalable Video Coding "Call for Proposals on Scalable Video Coding," JISO/IEC JTC1/SC29/WG11, MPEG2003/N5958, Brisbane, October 2003
- [4] A. Majumdar, J. Chou, K. Ramchandran, "Robust Distributed Video Compression based on Multilevel Coset Codes"
- [5] A. Sehgal and N. Ahuja, "Robust Predictive Coding and the Wyner-Ziv Problem," in *Proc. DCC*, 2003.
- [6] A. Aaron, S. Rane, R. Zhang, and B. Girod, "Wyner-Ziv Coding of Video: Applications to compression and error resilience," in *Proc. IEEE Data Compression Conf.*, 2003.
- [7] A. D. Wyner and J. Ziv, "The rate distortion function for source coding with side information at the decoder," *IEEE Trans. Inf. Theory*, vol. 22, pp. 1-10, Jan 1976.
- [8] S. S. Pradhan, J. Chou, and K. Ramchandran, "Duality between source coding and channel coding and its extension to the side information case" in *IEEE Trans. Inf. Theory*, vol. 49, May 2003.
- [9] P. Ishwar, V. M. Prabhakaran, and K. Ramchandran, "Towards a Theory for Video Coding Using Distributed Compression Principles," in *Proc. IEEE Int. Conf. Image Proc.*, 2003.
- [10] S. Yaman and G. AlRegib, "A Low Complexity Video Encoder with Decoder Motion Estimator," in *Proc. ICASSP*, 2004.



**Fig. 6.** Performance comparison of proposed scalable codec, H.264/AVC, and H.264/AVC protected with FECs (Reed-Solomon codes used, 20% of the total rate used for parity bits) for the Stefan sequence (CIF, 15fps, 1800 kbps).



**Fig. 7.** PSNR vs. frame number for the Football (CIF, 15fps, 1800 kbps) sequence at an error rate equal to 7.58%.

- [11] Q. Xu and Z. Xiong, "Layered Wyner-Ziv video coding," in *Proc. VCIP'04*, Jan. 2004.
- [12] A. Aaron and S. Rane and D. Rebello-Monedero and B. Girod, "Systematic Lossy Forward Error Protection for Video Waveforms," in *Proc. ICIP*, 2003.
- [13] "TIA/EIA Interim Standard for CDMA2000 Spread Spectrum Systems," May 2002.
- [14] S. S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (DISCUS): Design and construction," in *Proc. IEEE Data Compression Conf.*, 1999.
- [15] A.D. Liveris and Z. Xiong and C.N. Georghiades, "Distributed compression of binary sources using conventional parallel and serial concatenated convolutional codes," in *Proc. IEEE Data Compression Conf.*, 2003.
- [16] A. Aaron and B. Girod, "Compression with Side Information Using Turbo Codes," in *Proc. IEEE Data Compression Conf.*, 2002.
- [17] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: John Wiley and Sons, 1991.



(a) Proposed Codec: Base Layer only (QCIF)



(b) Proposed Codec: Base Layer and Enhancement Layer (CIF)



(c) H.264/AVC (CIF)

**Fig. 8.** Comparison of Frame 8 of the Stefan sequence (15fps, 1800 kbps) reconstructed by our proposed codec and H.264/AVC at a channel error rate equal to 8%. Here we have only shown the results for spatial scalability.