RATE-DISTORTION OPTIMIZED MERGE FRAME USING PIECEWISE CONSTANT FUNCTIONS

Wei Dai[o], *Gene Cheung*[#], *Ngai-Man Cheung*[*], *Antonio Ortega*[$], *Oscar C. Au*[o]

[o] Hong Kong University of Science and Technology, [#] National Institute of Informatics,
[*] Singapore University of Technology and Design, [$] University of Southern California

ABSTRACT

The ability to efficiently switch from one pre-encoded video stream to another is a valuable attribute for a variety of interactive streaming applications, such as switching among streams of the same video encoded in different bit-rates for real-time bandwidth adaptation, or view-switching among videos capturing the same dynamic 3D scene but from different viewpoints. It is well known that intra-coded I-frames can be used at switch boundaries to facilitate stream-switching. However, the size of an I-frame is large, making frequent insertion impractical. A recent proposal towards a more efficient stream-switching mechanism is distributed source coding (DSC), which exploits worst-case correlation between a set of potential predictor frames in the decoder buffer (called side information (SI) frames) and a target frame to lower encoding rate. However, the conventional use of bit-plane and channel coding means the encoding and decoding complexity of DSC frames is large. In this paper, we pursue a novel approach to the stream-switching problem based on the concept of "signal merging", using piecewise constant (pwc) function as the merge operator. Specifically, we propose a new merge mode for a code block, where for each k-th transform coefficient in the block, we encode appropriate step size and horizontal shift parameters at the encoder, so that the resulting floor function at the decoder can map corresponding coefficients from any SI frame to the same reconstructed value, resulting in an identically merged signal. The selection of shift parameter per coefficient, as well as coding modes between intra and merge per block, are optimized in a rate-distortion (RD) optimal manner. Experiments show encouraging coding gain over a previous implementation of DSC frame at low- to mid-bitrates at reduced computation complexity.

Index Terms— Interactive video, video coding, distributed source coding

1. INTRODUCTION

In conventional *non-interactive* video streaming, a client passively consumes every frame in a pre-encoded video stream as the video is played back in time in a fixed, rigid order. In contrast, in *interactive* video streaming [1], a client can in real-time freely choose subsets of a high-dimensional media content for personalized consumption. In response, the server must transmit pre-encoded data that corresponds to the requested media subsets for correct decoding and display at client. Examples of interactive video streaming include switching among streams of the same video encoded at different bit-rates for real-time bandwidth adaptation [2], static 3D scene navigation among light field images captured from different viewpoints [3, 4], view-switching among videos capturing the same dynamic 3D scene from different cameras [5], etc.

To support interactive video streaming, the technical challenge is to pre-encode a high-dimensional video content efficiently, while providing flexible mechanisms to facilitate stream-switching. One simple method is to insert an intra-coded I-frame at each application-required switching point. While I-frames, which do not require any predictor frame at decoder buffer for decoding, can facilitate stream-switching, their large size means frequent insertion is not practical.

Towards a more efficient stream-switching mechanism, *distributed source coding* (DSC) exploits the correlation between the set of possible frames from which a client is switching (called *side information* (SI)) and the target frame for coding gain [6, 7, 8]. Specifically, each code block is first mapped from pixel to transform domain. Then bit-planes of each transform coefficient from all SI frames are compared to the bit-planes of the target frame. The "noisiest" bit-planes among SI frames—ones with the largest deviation from target frame—are then identified, and channel codes strong enough to overcome the worst-case noise are encoded as the DSC frame. At the decoder, any SI frame *plus* DSC frame can result in an identical reconstruction of the target frame. DSC frame can potentially be much smaller in size than a comparable I-frame [8].

However, there remain significant problems in the DSC frame design. First, use of bit-plane encoding and channel codes means the computation complexity in both encoder and decoder is high. Further, because average statistics of a transform coefficient bit-plane for the entire image are used, non-stationary noise statistics can lead to high rate channel codes, resulting in coding inefficiency.

In this paper, we pursue a novel approach to the stream-switching problem based on the concept of "signal merging"—*merging any SI into an identically reconstructed good signal*—using piecewise constant (pwc) function as the merge operator. Specifically, we propose a new merge mode for a code block, where for the k-th transform coefficient in the block, we encode appropriate step size and horizontal shift parameters of a `floor` function at the encoder, so that the resulting `floor` function at the decoder can map corresponding coefficients from any SI frame to the same reconstructed value. The selection of step size and horizontal shift directly affects both the merged signal fidelity and the coding rate; we propose rate-distortion (RD) optimization procedures to optimize these parameters, as well as the selection of coding modes between intra and merge on a per block basis. Experimental results show encouraging coding gain over a previous implementation of DSC frame at low- to mid-bitrates at reduced computation complexity.

The outline of the paper is as follows. We first outline related work in Section 2. We then overview our coding system in Section 3. We discuss the use of pwc functions for signal merging in Section 4, and propose simple RD optimization procedures in Section 5. Finally, we present experimental results and conclusions in Section 6 and 7, respectively.

2. RELATED WORK

Beyond conventional I-, P- and B-frames, H.264 [9] introduced *SP-frames* [10] for stream-switching. In a nutshell, first a primary SP-frame is differentially coded between one predictor frame and the target frame, similar to a conventional P-frame. Then, for each additional predictor frame, one secondary SP-frame is differentially coded between the predictor frame and the reconstructed primary SP-frame, where the prediction residual is losslessly coded to ensure identical reconstruction between primary and secondary SP-frames. Due to the lossless coding employed, secondary SP-frames are coding-inefficient and are inferior to DSC frames [8].

While DSC has been a popular concept in designing stream-switching mechanisms in the past decade [2, 3, 6, 7, 8], partly due to the computation complexity required for bit-plane and channel coding, DSC is neither widely used nor adopted into any video coding standards. In contrast, our implementation of merge frame based on the concept of "signal merging" involves only quantization (pwc function) and arithmetic coding of horizontal shifts, both of which are simple, intuitive and well understood in the coding community.

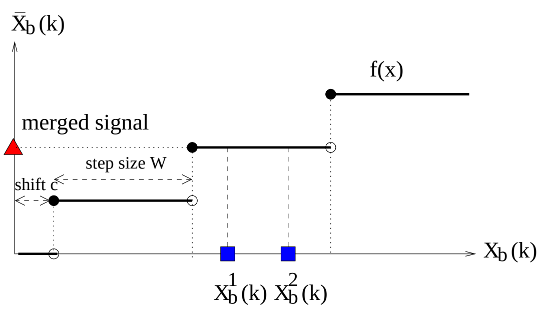


Fig. 1. Given the side information $X_b^1(k)$ and $X_b^2(k)$ corresponding to the k-th coefficient of block b, a pwc function $f(x)$ maps them to the same $\bar{X}_b(k)$ if they fall on the same step. Only step size W and shift c are needed to specify pwc function $f(x)$, which we encode for each coefficient.

There exists an interesting connection between our concept of "signal merging" and classical *coset coding* in DSC [11]. Coset coding can be interpreted as extraction of *coarse*-grained information from noise-corrupted SI, plus transmission of *fine*-grained information—also known as coset indices—to recover the original signal *exactly*. On the other hand, the requirement for signal merging—identical reconstruction of a merged signal given any one SI plus transmitted bits—is less stringent. As an example, in Fig. 1 we see that the k-th transform coefficients of block b from two SI frames, $X_b^1(k)$ and $X_b^2(k)$, fall on the same step of pwc function $f(x)$, meaning either one of the two values can be mapped via $f(x)$ to the same (merged) signal $\bar{X}_b(k)$. Clearly, given $X_b^1(k)$ and $X_b^2(k)$ are both known during encoding, encoder can choose among many combinations of step size W and shift c to guarantee unique reconstruction, including combos that recover the target frame's original k-th coefficient $Y_b(k)$ exactly. Unlike coset coding, however, encoder can use this degree of freedom to choose W and c that optimally trade off rate and distortion. In particular, in Section 5.2 we will argue how our selection of shifts c's can lead to favorable statistics for arithmetic coding of shift parameters.

3. SYSTEM OVERVIEW

We overview our proposed coding system in which a *merge frame*—where code blocks can be coded using our designed merge mode

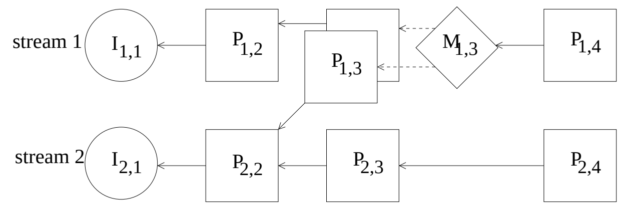


Fig. 2. Example of stream-switching from one pre-encoded stream to another using merge frame. SI frames ($P_{1,3}$'s) are first constructed using different predictors. Then merge frame $M_{1,3}$ is encoded using the two SI frames. I-, P- and merge frames are represented as circles, squares and diamonds.

or intra mode—is used as a stream-switching mechanism. First, for each possible switch from frame F_i to frame F_j, an *SI frame* that is a P-frame differentially coded using F_i as predictor and F_j as target is encoded. Thus the SI frames constitute the best approximation of the target frame given their respective predictor frames. In Fig. 2, two P-frames $P_{1,3}$'s of stream 1 and time instant 3, predicted from predictor frames $P_{1,2}$ and $P_{2,2}$ of streams 1 and 2, represent the SI frames. Then a merge frame ($M_{1,3}$ in Fig. 2) is encoded to merge any possible SI frame to an identically constructed version of the target frame. During a stream-switch, the server can transmit any one of the SI frames *plus* the merge frame for an identical reconstruction, and avoid coding drift in the following frames that predict from the merge frame. The challenge is to design a merge frame in this setting in an RD optimal manner. We describe this next.

4. PWC FUNCTIONS FOR SIGNAL MERGING

Let the N SI frames be $\mathbf{S}^1, \ldots, \mathbf{S}^N$. There is an uncertainty at encoding time as to which one of these N SI frames will be available at decoder buffer for decoding of the merge frame, but the set of N SI frames is known at encoding time with certainty. Let the reconstructed image after decoding the merge frame be $\bar{\mathbf{T}}$, which is an approximation of the target image $\mathbf{T}$. The goal is to design the merge frame $\mathbf{M}$ such that distortion with respect to target image $\mathbf{T}$, $D_{\mathbf{T}}(\mathbf{M})$, and the encoding rate of the merge frame, $R(\mathbf{M})$, are optimally traded off:

$$\min_{\mathbf{M}} D_{\mathbf{T}}(\mathbf{M}) + \lambda R(\mathbf{M}) \tag{1}$$

where each combination of SI frame $\mathbf{S}^n$ and merge frame $\mathbf{M}$ can *identically* reconstruct to $\bar{\mathbf{T}}$. The goal is to find the best merge frame $\mathbf{M}$ possible in an RD-optimal sense.

We first discuss a framework for merge frame construction using pwc function. We then discuss the resulting distortion and rate costs for different choices of parameters in the framework.

4.1. Piecewise Constant Function for Single Merging

Suppose a K-pixel code block of index b from SI frame $\mathbf{S}^n$, $\mathbf{x}_b^n$, is transformed and quantized to $\mathbf{X}_b^n = [X_b^n(0), \ldots, X_b^n(K-1)]$, where $X_b^n(k)$ is the quantization index of k-th coefficient of block b of SI frame n. (This implies $X_b^n(k)$ is an integer, i.e., $X_b^n(k) \in \mathbb{I}$.) To have identical reconstruction $\bar{\mathbf{X}}_b$ for block b in reconstructed frame $\bar{\mathbf{T}}$, each one of k-th coefficients $X_b^1(k), \ldots, X_b^N(k)$ must map to the same quantization index $\bar{X}_b(k)$. This can be accomplished through a pwc function, as illustrated in Fig. 1 for $N = 2$. Examples of pwc functions are `mod`, `round`, `floor`, etc. We will restrict our attention to the `floor` function[1]:

$$f(x) = \left\lfloor \frac{x+c}{W} \right\rfloor W + \frac{W}{2} - c \qquad (2)$$

where the *step size* is W and the *horizontal shift* is c.

We know that each SI frame is correlated with the target frame, which would imply that the SI frames themselves are correlated. Hence, the largest difference between any pair in $X_b^1(k), \ldots, X_b^N(k)$ for k-th coefficient in block b is small on average. Let $W_b(k)$ be the maximum difference between two k-th coefficients in block b from any two SIs, i.e.

$$W_b(k) = \max_{n=1,\ldots,N} X_b^n(k) - \min_{n=1,\ldots,N} X_b^n(k) \qquad (3)$$

Given $W_b(k)$, we can next define group-wise maximum difference for a group $\mathcal{B}$ of blocks, $W_{\mathcal{B}}(k)$:

$$W_{\mathcal{B}}(k) = \max_{b \in \mathcal{B}} W_b(k) \qquad (4)$$

Given $X_b^n(k)$'s are integers, $W_{\mathcal{B}}(k)$ is also an integer.

For any block b in group $\mathcal{B}$, a step size $W_{\mathcal{B}}^+(k) = W_{\mathcal{B}}(k) + \epsilon$ is sufficient for `floor` function $f(x)$ to map any coefficient in $X_b^1(k), \ldots, X_b^N(k)$ to the same value, for any $\epsilon > 0$, *if* horizontal shift $c_b(k)$ can be appropriately chosen. In order to use the smallest step size $W_{\mathcal{B}}^+(k)$ possible while keeping $c \in \mathbb{I}$, we fix ϵ to be 1.

Shift $c_b(k)$ must be chosen such that any coefficient in $X_b^n(k)$, $n \in \{1, \ldots, N\}$, is mapped to the same value via $f(x)$, i.e.:

$$\left\lfloor \frac{X_b^1(k) + c_b(k)}{W_{\mathcal{B}}^+(k)} \right\rfloor = \left\lfloor \frac{X_b^n(k) + c_b(k)}{W_{\mathcal{B}}^+(k)} \right\rfloor \quad \forall n \in \{1, \ldots, N\}$$

One can show that feasible set of values of $c_b(k) \in \mathbb{I}$, denoted as $\mathcal{F}_b(k)$, are:

$$c_b^{\min}(k) + mW_{\mathcal{B}}^+(k) \leq c_b(k) \leq c_b^{\max}(k) + mW_{\mathcal{B}}^+(k) \qquad (5)$$

where m is an integer, and $c_b^{\min}(k)$ and $c_b^{\max}(k)$ are defined as:

$$\begin{aligned} c_b^{\min}(k) &= -(X_b^{\min}(k) \;\texttt{mod}\; W_{\mathcal{B}}^+(k)) & (6)\\ c_b^{\max}(k) &= c_b^{\min}(k) + W_{\mathcal{B}}(k) - W_b(k) & (7) \end{aligned}$$

Note that (4) implies $W_{\mathcal{B}}(k) \geq W_b(k)$, and so $c_b^{\max}(k) \geq c_b^{\min}(k)$. Note also that $c_b^{\max}(k), c_b^{\min}(k) \in \mathbb{I}$.

4.2. Distortion Cost

Different horizontal shifts $c_b(k)$'s in feasible set $\mathcal{F}_b(k)$ (5) induce different distortion in the reconstructed signal. We first define distortion for k-th coefficient of block b, $d_b(k)$, to be the difference between original k-th coefficient $Y_b(k)$ of the target image $\mathbf{T}$ and reconstructed coefficient $f(X_b^1(k))$:

$$d_b(k) = |Y_b(k) - f(X_b^1(k))|^2 \qquad (8)$$

Because we assume a valid horizontal shift $c_b(k)$ is chosen, all N k-th coefficients $X_b^n(k)$'s map to the same value $f(X_b^n(k)), \forall n \in \{1, \ldots, N\}$. Thus we consider only $f(X_b^1(k))$ in (8).

Given $f(x)$ in (2) and step size W, shift c, $c \in \mathbb{I}$, is capable of moving input x only within a neighborhood of W integers around x. To see this, let $x = x_1 W + x_2$, where $x_1, x_2 \in \mathbb{I}$ and $0 \leq x_2 \leq W - 1$. Similarly, we can write $c = c_1 W + c_2$. Assume first the case where $x_2 + c_2 < W$. (2) can be now analyzed as:

$$\begin{aligned} f(x) &= \left\lfloor \frac{(x_1 + c_1)W + (x_2 + c_2)}{W} \right\rfloor W + \frac{W}{2} - c \\ &= (x_1 + c_1)W + \frac{W}{2} - c_1 W - c_2 \\ &= x_1 W + \frac{W}{2} - c_2 \end{aligned}$$

Hence c can *decrease* $f(x)$ by c_2, given $0 \leq c_2 < W - x_2$.

Now consider the case where $x_2 + c_2 \geq W$. Let $W + b_2 = x_2 + c_2$, where $0 \leq b_2 \leq W - 2$. (2) becomes:

$$\begin{aligned} f(x) &= \left\lfloor \frac{(x_1 + c_1 + 1)W + b_2}{W} \right\rfloor W + \frac{W}{2} - c \\ &= (x_1 + c_1 + 1)W + \frac{W}{2} - c_1 W - c_2 \\ &= (x_1 + 1)W + \frac{W}{2} - c_2 \end{aligned}$$

Hence c can *increase* $f(x)$ by $W - c_2$, where $W - x_2 \leq c_2 \leq W - 1$.

We can make two observations from the above analysis. First, though the feasible set $\mathcal{F}_b(k)$ for $c_b(k)$ as described in (5) is large, a restricted range of $0 \leq c_b(k) \leq W - 1$ is sufficient to induce all possible changes in $f(x)$. Second, the larger the step size W, the larger the range of values $c_b(k)$ can augment in $f(x)$ (though the resulting coding cost of $c_b(k)$ will also increase, to be discussed next).

4.3. Coding Cost

Using `floor` function $f(x)$ for transform coefficient merging, we can thus conclude that the encoding cost for the k-th coefficient in block group $\mathcal{B}$ of the merge frame $\mathbf{M}$ is the following:

1. one step size $W_{\mathcal{B}}^+(k) = W_{\mathcal{B}}(k) + 1$ for group $\mathcal{B}$.
2. one horizontal shift $c_b(k)$ for each block in group $\mathcal{B}$.

The cost of encoding a single $W_{\mathcal{B}}(k)$ for k-th coefficients of a large group $\mathcal{B}$ is small. The cost of encoding $|\mathcal{B}|$ horizontal shifts $c_b(k)$'s for k-th coefficient, on the other hand, can be expensive. We hence focus our next discussion on two important design components for good RD performance: i) identification of blocks in a frame as merge blocks in merge group $\mathcal{B}$, and ii) efficient coding of horizontal shifts $c_b(k)$'s for blocks in group $\mathcal{B}$.

5. RD OPTIMIZATION

5.1. RD-optimal Selection of Block Modes

For blocks that are very different across SI frames, they will require too many bits to code parameters of the `floor` function for signal merging. We perform the following procedure to identify blocks that should be coded as intra blocks instead. First, we encode blocks of the entire frame as merge blocks, resulting in a certain average distortion per block $\hat{d}$. We then select the quantization parameter (QP) for intra-coded blocks, so that the average distortion is also $\hat{d}$.

For each block b in a snake order, we evaluate its RD cost when coding using mode m: $D_b(m) + \lambda R_b(m)$, where $m \in \{\texttt{intra}, \texttt{merge}\}$. For `intra`, the intra-coded block's distortion D_b and rate R_b are determined by the aforementioned QP. Distortion and rate for a block coded in `merge` mode are the corresponding

[1] `floor` function is defined here such that the *maximum* difference between original x and reconstructed $f(x)$, after horizontal shift of c and `floor` operation with step size W, is minimized.

sums of distortion and rate for individual frequencies in the block, which is described in the next section. The mode with the smaller RD cost is selected for encoding of block b.

Given the blocks chosen for coding in `merge`, let $W^*(k) = \max_b W_b(k)$ be the largest $W_b(k)$ of k-th coefficient in these blocks. We then re-optimize the shift selection in these blocks given $W^*(k)$'s. Note that because the blocks with large difference among SI frames have been declared intra blocks, the remaining blocks should be more similar, which will result in smaller $W^*(k)$'s. Small step size $W^*(k) + 1$ means the size of the alphabet for encoding of horizontal shifts $c_b(k)$, $c_b(k) \in [0, W^*(k))$, is also small, leading to coding gain.

5.2. RD-optimal Selection of Horizontal Shifts

Given chosen $W^*(k)$ as described previously, we reselect horizontal shift $c_b(k)$ for k-th coefficient of each block b using the following RD criteria:

$$\min_{0 \leq c \leq W^*(k) \mid c \in \mathcal{F}_b(k)} d_b(k) + \lambda(-\log Pr(c - c_{b-1}(k))) \tag{9}$$

where $d_b(k)$ is the distortion defined in (8), and the rate term is the negative log of the probability of the difference between the current $c_b(k)$ and previous $c_{b-1}(k)$ for previous block $b-1$. In other words, we code only the difference in shift $\Delta_b(k) = c_b(k) - c_{b-1}(k)$ between current and previous blocks. $\mathcal{F}_b(k)$ is the feasible set for $c_b(k)$ given step size $W^*(k) + 1$.

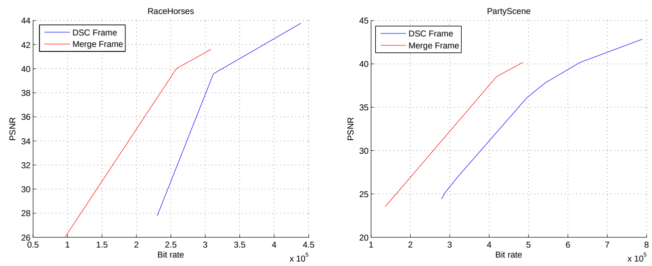
In our implementation, the shift differentials $\Delta_b(k)$'s for difference frequencies k in a given block b are coded together as one codeword using arithmetic coding (AC). Starting with an initial probability distribution for $\Delta_1(k)$ for frequency k of the first block, each subsequent block derives an updated distribution based on collected statistics of previous coded merge blocks for this frequency. Further, given high frequency components of the target frame are likely zero or close to zero, we encode an End-of-Block (EoB) symbol when the remaining frequency components of the target block are all smaller than a threshold ρ. This means we only need to encode a small number of shift differentials $\Delta_b(k)$'s per block.

We argue that when using (9) to select shifts $c_b(k)$'s for large λ value, the resulting statistics for $\Delta_b(k)$'s can be much more favorable for compression than coset indices in coset coding [11]. Given a single $W^*(k)$ is chosen for k-th coefficients of all merge blocks in the frame, it is likely much larger than the typical maximum coefficient difference $W_b(k)$ among SI frames for many blocks b's. A large relative step size in the pwc function means that the same shift $c_b(k)$ can likely be reused for signal merging for a long sequence of blocks, i.e., $\Delta_b(k) = 0$ has high probability, resulting compression gain when using AC. This is in contrast to coset indices in coset coding, whose statistical behavior is more similar to least significant bits (LSB), which is very random. The cost of choosing $\Delta_b(k) = 0$ often in (9) is a penalty in distortion. As we will see in the results section, our merge frame shows more promise at low bitrate region.

6. EXPERIMENTATION

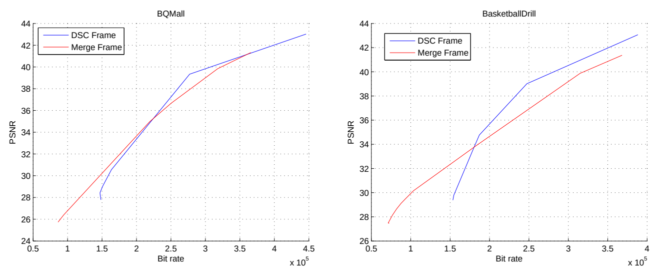
To test the performance of our proposed merge frame, we conducted the following experiment. We used the following four video test sequences: `RaceHorses`, `PartyScene`, `BQMall` and `BasketballDrill`[2]. For each sequence, we first prepared two SI frames, which were predictively coded using as predictors different coded versions of the previous frame compressed with

[2] ftp://ftp.tnt.uni-hannover.de/testsequences/



(a) `RaceHorses` (b) `PartyScene`

Fig. 3. PSNR versus encoding rate comparing proposed merge frame with DSC frame in [6] for sequences `RaceHorses` and `PartyScene`.



(a) `BQMall` (b) `BasketballDrill`

Fig. 4. PSNR versus encoding rate comparing proposed merge frame with DSC frame in [6] for sequences `BQMall` and `BasketballDrill`.

different QP. Then, QP for the SI frames themselves were varied to induce different RD tradeoff. Given a pair of SI frames encoded at a particular QP, we encoded our merge frames using a range of λ values when choosing horizontal shifts c using (9). The convex hull of all operational points represents the RD performance of our proposed merge frame. For comparison, we also plotted the RD performance of an earlier implementation of DSC scheme in [6].

In Fig. 3, we see the RD performance for sequences `RaceHorses` and `PartyScene`. We see that our proposed merge frame outperformed DSC frame in [6] at all bitrate regions. In Fig. 4, we see the RD performance for sequences `BQMall` and `BasketballDrill`. For these two sequences, we see that our proposed merge frame outperformed DSC frame in [6] only at low- to mid-bitrate regions. Though not shown, we observed that statistics for $\Delta_b(k)$'s are more skewed towards $\Delta_b(k) = 0$ at low bitrate for all sequences, meaning horizontal shifts have more favorable statistics at low bitrate for coding using AC, which explains the performance gain.

7. CONCLUSION

Designing a stream-switching mechanism that is also coding-efficient is a difficult task. In this paper, we pursue a novel approach based on the concept of “signal merging”, using piecewise constant (pwc) function as the merging operator. Specifically, in order to merge k-th transform coefficients of different side information (SI) frames to the same value, we encode appropriate step size and horizontal shift parameters of a `floor` function, so that all the SI coefficients fall on the same function step. We propose RD optimization techniques to select shift parameters for each coefficient, as well as coding mode between intra and merge for each block. Experimental results show encouraging coding gain over a previous implementation of DSC frame at low- to mid-bitrates at much reduced computation complexity.

8. REFERENCES

[1] G. Cheung, A. Ortega, N.-M. Cheung, and B. Girod, "On media data structures for interactive streaming in immersive applications," in *SPIE Visual Communications and Image Processing*, Huang Shan, China, July 2010.

[2] M. Guo, Y. Lu, F. Wu, D. Zhao, and W. Gao, "Wyner-Ziv switching scheme for multiple bit-rate video streaming," in *IEEE Transactions on Circuits and Systems for Video Technology*, May 2008, vol. 18, no.5, pp. 569–581.

[3] A. Aaron, P. Ramanathan, and Bernd Girod, "Wyner-Ziv coding of light fields for random access," in *IEEE International Workshop on Multimedia Signal Processing*, Siena, Italy, September 2004.

[4] P. Ramanathan and B. Girod, "Random access for compressed light fields using multiple representations," in *IEEE International Workshop on Multimedia Signal Processing*, Siena, Italy, September 2004.

[5] G. Cheung, A. Ortega, and N.-M. Cheung, "Interactive streaming of stored multiview video using redundant frame structures," in *IEEE Transactions on Image Processing*, March 2011, vol. 20, no.3, pp. 744–761.

[6] N.-M. Cheung, H. Wang, and A. Ortega, "Video compression with flexible playback order based on distributed source coding," in *IS&T/SPIE Visual Communications and Image Processing (VCIP'06)*, San Jose, CA, January 2006.

[7] N.-M. Cheung and A. Ortega, "Compression algorithms for flexible video decoding," in *IS&T/SPIE Visual Communications and Image Processing (VCIP'08)*, San Jose, CA, January 2008.

[8] N.-M. Cheung, A. Ortega, and G. Cheung, "Distributed source coding techniques for interactive multiview video streaming," in *27th Picture Coding Symposium*, Chicago, IL, May 2009.

[9] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," in *IEEE Transactions on Circuits and Systems for Video Technology*, July 2003, vol. 13, no.7, pp. 560–576.

[10] M. Karczewicz and R. Kurceren, "The SP- and SI-frames design for H.264/AVC," in *IEEE Transactions on Circuits and Systems for Video Technology*, July 2003, vol. 13, no.7, pp. 637–644.

[11] S. S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (DISCUS): design and construction," in *IEEE Transactions on Information Theory*, March 2003, vol. 49, no.3, pp. 626–643.