Drift-Free Switching of Compressed Video Bitstreams at Predictive Frames

Xiaoyan Sun, Feng Wu, Shipeng Li, Guobin Shen, *Member, IEEE*, and Wen Gao

***Abstract*—Two schemes are proposed to efficiently compress video contents into bitstreams that support drift-free switching at predictive frames. They are inspired by the original SP coding scheme presented in the early H.26L. First, we propose a Flex SP coding scheme in which the prediction signal of the SP frame is directly subtracted from the input without quantization and de-quantization. The decoded video quality of the Flex SP scheme is significantly improved when additional inverse discrete cosine transform (DCT) and post-filter are provided. Then, the Hybrid SP scheme is presented to further improve the quality of the display image, as well as the reconstructed reference, by defining two coding modes for each DCT coefficient. Moreover, a rate-distortion algorithm is proposed to determine the coding mode for each coefficient. The bitstreams generated by the two proposed schemes can be decoded successfully by a decoder that complies with MPEG-4 AVC/H.264. In addition, we also investigate how to choose the quantization parameters for switching. An empirical method is proposed to achieve a good tradeoff between high coding efficiency of SP frames and small size of switching bits.**

***Index Terms*—Bitstream switching, multiple-bit-rate (MBR) streaming, rate-distortion (R-D) optimization, SP frame, video coding.**

I. Introduction

TO COPE WITH network bandwidth variations, the multiple-bit-rate (MBR) method is often used for streaming video contents over the Internet, and it has been extensively supported by commercial streaming products, such as the Windows Media system, RealSystem, and QuickTime [1]–[3]. The switching among multiple bitstreams at different bit rates is normally accomplished by inserting some special frames in the bitstreams, known as *key frames*, which are independently compressed without temporal prediction [4].

It is well known that I-frames can be used for the purpose of switching between bitstreams since no temporal prediction is involved. However, an I-frame requires many more bits than a P-frame does at the same decoded quality. Thus, the coding efficiency will be greatly reduced when I-frames are frequently inserted to support rapid switching between bitstreams. On the other hand, a P-frame is always coded based on forward prediction with reference to a previous I- or P-frame. Therefore, if bitstreams were switched at P-frames without any special treatment, it would result in visual artifacts due to the mismatch of the reconstructed references at the switching point. Moreover, the mismatch errors would propagate and accumulate in the subsequent P-frames until another I-frame is reached. Such errors are often referred to as *drifting errors*. This could rapidly deteriorate the decoded visual quality as the number of P-frames increases.

Manuscript received November 28, 2003; revised September 29, 2005. This work was supported in part by the Natural Science Foundation of China under Grant 60333020. This paper was recommended by Associate Editor A. Tabatabai.

X. Sun was with the Harbin Institute of Technology, Harbin, 150001, China. She is now with Microsoft Research Asia, Beijing 100081, China (e-mail: xysun@microsoft.com).

F. Wu, S. Li, and G. Shen are with Microsoft Research Asia, Beijing 100081, China (e-mail: fengwu@microsoft.com; spli@microsoft.com; jackysh@microsoft.com).

W. Gao is with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100085, China (e-mail: wgao@jdl.ac.cn).

Digital Object Identifier 10.1109/TCSVT.2006.873162

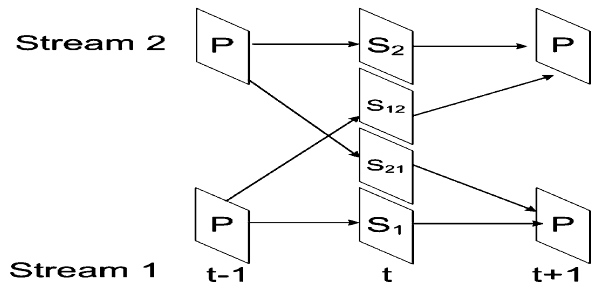


Fig. 1. Switching between Streams 1 and 2 through SP frames.

Farber *et al.* first proposed a special predictive frame, called the S-frame, to achieve switching among nonscalable bitstreams at predictive frames [5]. However, it was shown that S-frames tend to drift, and, in order to keep drifting error within a small scale, the size of the S-frame has to be considerably large [19]. On the other hand, a technique which supports drift-free switching at predictive frames has been proposed to the JVT standard in [6] and [7] and accepted as a new picture type, called the SP frame [9]. Similar to P-frames, SP frames exploit temporal redundancy by motion-compensated predictive coding. Unlike P-frames, SP frames enable identical reconstruction of the target frame even when different reference frames are used. This empowers drift-free switching from one bitstream to another at predictive frames.

Fig. 1 illustrates the process of switching between bitstreams through SP frames. Two bitstreams, Streams 1 and 2, are generated at different bit rates. Assume that the bit rate of Stream 1 is lower than that of Stream 2, the switching from Stream 1 to Stream 2 is referred to as *up-switching*, and the reverse is referred to as *down-switching* hereafter in this paper. The frames at $t-1$ and $t+1$ are coded as normal P-frames in both Streams 1 and 2. S_1, S_2, S_{12}, and S_{21} at time t are coded as SP frames, where S_{12} is used for switching from Stream 1 to Stream 2 and S_{21} is used for switching from Stream 2 to Stream 1. Although they are of the same picture type, in this paper, we would like to refer to S_1 and S_2 as *primary* SP frames and to S_{12} and S_{21} as *secondary* SP frames for convenience. Because of the temporal prediction, SP frames offer a more efficient way to switch

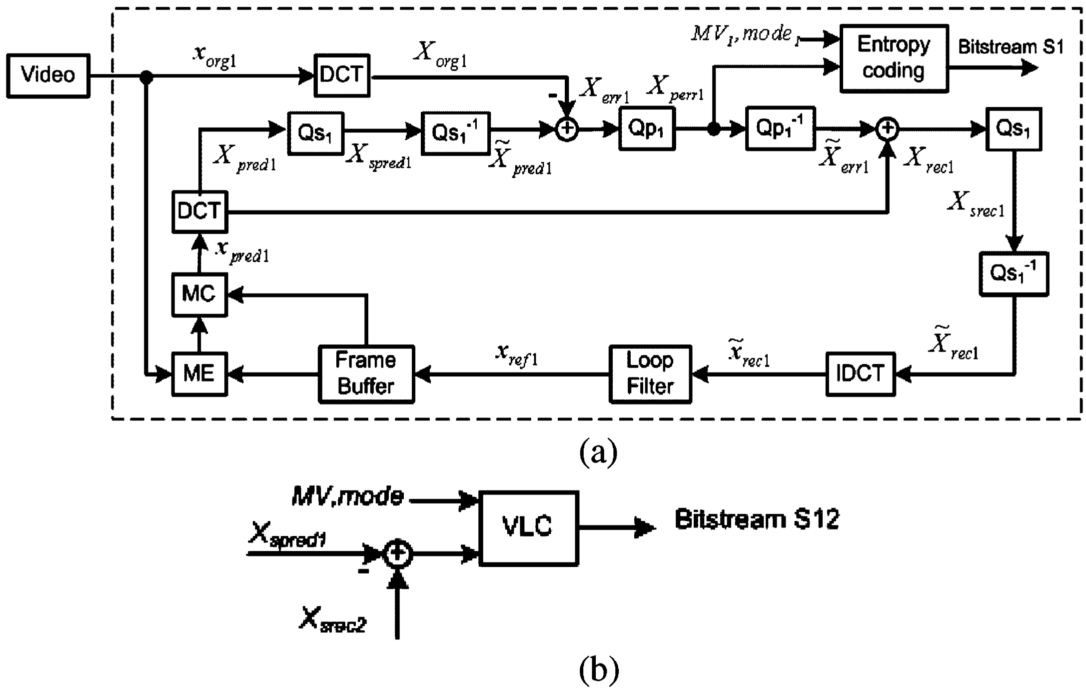


Fig. 2. Encoder of a common SP scheme. (a) Encoder of primary SP frame (S_1). (b) Encoder of secondary SP frame S_{12} (from S_1 to S_2).

between bitstreams than I-frames do. Moreover, SP frames provide extra functionalities for error resilience/recovery, random access, and fast-forward [6], [18].

Fig. 2 depicts the block diagram of a common encoder consistent with H.264. For simplicity, only the encoder of S_1 and S_{12} are shown here. The encoder of S_2 can be readily achieved by changing the subscript number in Fig. 2(a) from 1 to 2. Here, *Qp* is the quantization module as same as that in normal P-frame, while *Qs* and inverse *Qs* (Qs^{-1}) are additional quantization modules in SP frames. As shown in Fig. 1, at switching point t, S_2 and S_{12} are predicted from references in Streams 1 and 2, respectively. Then, the difference between the reconstruction of Stream 2 and the temporal prediction of Stream 1 would be coded losslessly in S_{12} for error-free switching. However, in this case, the size of S_{12} is too huge to be acceptable in many applications. Therefore, in SP-frame coding, the reconstructed frame of Stream 2 X_{rec2} and the temporal prediction of Stream $1 X_{\text{pred1}}$ are quantized with the same *Qs* step before generating the bitstream S_{12}, as shown in Fig. 2.

This paper proposes two schemes to improve SP-frame encoding in terms of coding performance and flexibility of bitstream switching. The Flex SP scheme enables high-quality images for display before the quantizer *Qs* in the reconstruction loop. By decoupling the quantizer *Qs* in two switching bitstreams, it allows independent quantization parameters as well as separated switching points for up-switching and down-switching. As a result, the size of the down-switching bitstream can be encoded much smaller than that of the up-switching bitstream, and the Flex SP scheme can readily support more rapid and more frequent down-switching than up-switching. Such support is crucial for the MBR streaming video to provide a seamless user experience over a dynamic network. Moreover, the two normal bitstreams, Streams 1 and 2, can also be independently optimized to achieve good coding performance. The Flex SP scheme was proposed to the JVT standard in [10] and was also described in [11].

Despite the above advantages, the Flex SP scheme may not provide constant coding efficiency gain when the quantized reconstruction is output as display video for complexity-limited applications. This is because the quantized reconstructed reference may benefit from the same quantized prediction rather than the unquantized one. Therefore, the Hybrid SP coding scheme is proposed by defining two coding modes for each coefficient. This leads to a better prediction that improves the quality of both the reconstructed reference and the display image. Furthermore, a rate-distortion (R-D) algorithm is proposed in this paper to optimally select the temporal prediction mode for each coefficient. The proposal for the Hybrid SP coding scheme was submitted to the JVT standard in [13] and has been adopted by the JVT standard in [15]. Therefore, the two SP schemes presented in this paper completely conform to the final JVT standard.

Another interesting topic in SP-frame coding is how to choose the quantization parameter *Qs*. Obviously, for smaller *Qs*, the coding efficiency loss of primary SP frames is small, but the size of each secondary SP frame becomes large. It means that more bits have to be delivered at each switching point. On the other hand, a larger *Qs* will result in a lower coding efficiency of primary SP frames and a smaller size of each secondary SP frame. This indicates that, in the case of infrequent switching, the primary SP streams would suffer from a big coding efficiency loss without any compensation. In this paper, we analyze the joint R-D performance of primary SP frames and secondary SP frames and determine how to choose the parameter to achieve a good tradeoff between the coding performance of SP frames and the size of the switching bits.

This paper is organized as follows. Section II gives a detailed description of the encoder and decoder of the Flex SP scheme. Section III proposes the Hybrid SP scheme. The R-D algorithm

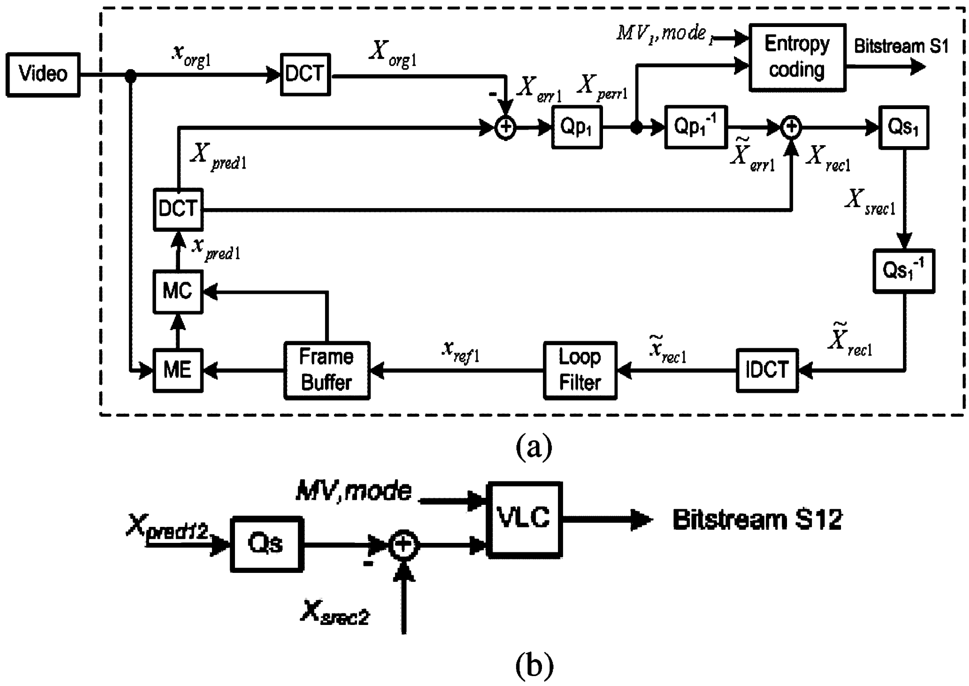


Fig. 3. Encoder of the Flex SP scheme. (a) Encoder of primary SP frame (S_1). (b) Encoder of secondary SP frame (from S_1 to S_2).

for optimally selecting the coding mode for each discrete cosine transform (DCT) coefficient is the focus of this section. The performance of the SP coding schemes is evaluated in Section IV. Section V discusses how to choose the switching quantization parameter. Finally, Section VI concludes this paper.

II. Flex SP Coding Scheme

The Flex SP coding scheme is discussed here. As mentioned earlier, for SP frames, an additional quantization *Qs* on the reconstructed frame is required to reduce the size of the switching bitstream, and the difference between the two quantized signals has to be coded by the secondary SP encoder to achieve drift-free switching. However, we observed that the first quantizer *Qs* operated on the temporal prediction in the primary SP frame encoder [shown in Fig. 2(a)] can be moved from the motion compensation loop into the secondary SP frame encoder without affecting the drift-free switching capability. There are some significant advantages of such a modification. First, if an SP frame is not a target frame to which some other bitstreams will switch, all additional quantizers *Qs* can be removed from the SP frame. Thus, it essentially becomes a normal P-frame. Second, if an SP frame is a target frame to which some other bitstreams will switch, the drift-free switching can always be supported as long as the quantizer *Qs* is in the secondary SP frame encoder.

To facilitate the description, we use the terms with two subscripts to denote the signal at each processing stage, and they are specified as follows.

First subscript (optional, only related to quantization):

- p: Quantization parameter specified by *Qp*;
- s: Quantization parameter specified by *Qs*.

Second subscript:

- Second subscript
- *org*: original signal;
- *pred*: predicted signal;
- *rec*: reconstructed signal;
- *ref*: reconstructed reference signal;
- *err*: predicted error signal.

Moreover, in the subsequent figures and descriptions, the uppercase letter X denotes the information in DCT domain, and the lowercase letter x denotes the information in pixel domain. The hat "~" represents the de-quantized signal. For example, X_{perr} means the quantized predicted error in the DCT domain with quantizer *Qp*, and $\tilde{x}_{\mathrm{err}}$ is the de-quantized predicted error in the pixel domain.

The encoder and decoder of the Flex SP scheme are illustrated in Figs. 3 and 4, respectively. Accordingly, the encoding of the primary SP frame [using S_1 as an example, as shown in Fig. 3(a)] is performed as follows. The original frame as well as the temporal prediction is DCT-transformed to generate X_{pred1} and X_{org1}. The difference between X_{pred1} and X_{org1} is quantized with Qp_1, and the obtained quantized error X_{perr1} is compressed into S_1 by entropy coding (CAVLC or CABAC). In addition, X_{perr1} is de-quantized, and the resulting de-quantized error $\tilde{X}_{\mathrm{err1}}$ is added to X_{pred1} to generate the DCT reconstruction X_{rec1}. Then, X_{rec1} is quantized and de-quantized with Qs_1. Finally, the reconstructed signal $\tilde{X}_{\mathrm{rec1}}$ is inverse DCT-transformed followed by a loop filter to provide the reconstructed reference x_{ref1}, which is used to update the frame buffer.

The encoding of the secondary SP frame S_{12} is shown in Fig. 3(b). The input X_{pred12} can be obtained in two ways. First, the prediction X_{pred1} in S_1 or a normal P-frame is directly used as X_{pred12}. This is applicable to switching between two weakly correlated bitstreams. Second, X_{pred12} is different from X_{pred1} since new motion estimation and compensation referencing the previous frame in Stream S_1 is preformed and the

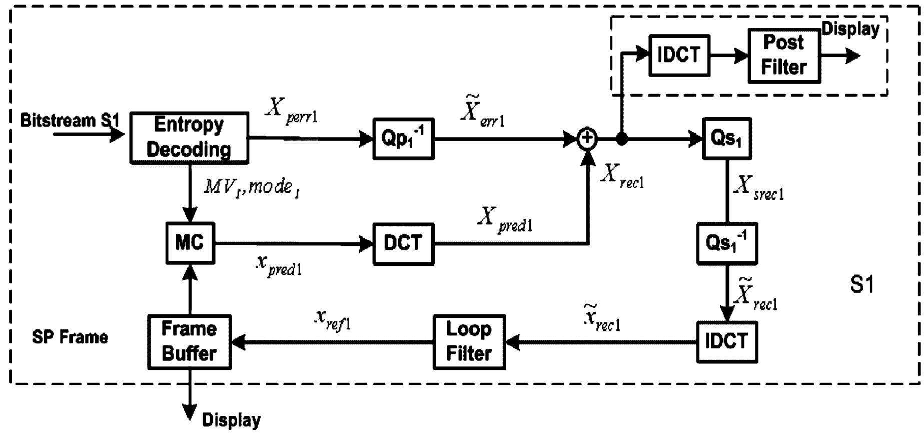


Fig. 4. Decoder of the Flex SP scheme.

resulting prediction is transformed to provide the DCT prediction X_{pred12}. As this method tries to reduce the temporal redundancy between bitstreams, it is suitable to code two correlated bitstreams. In any event, X_{pred12} is quantized with Qs_2 subsequently, and the resulting quantized prediction X_{spred12} is subtracted from X_{pred12} (provided by the target primary SP frame encoder). Eventually, the prediction error X_{err12} is compressed into Stream S_{12} by entropy coding (CAVLC or CABAC).

The decoding of the primary SP frame is shown in Fig. 4. Given S_1 as an example, after entropy decoding of the Stream S_1, the quantized error X_{perr1} and macroblock modes and motion vectors are obtained. Then, X_{perr1} is de-quantized with Qp_1, and the resulting residue is added to the DCT prediction X_{pred1} to generate the reconstruction x_{rec1}. x_{rec1} goes through the optional inverse DCT transform and de-blocking filter to form a image for display. Meanwhile, X_{rec1} is quantized and de-quantized with Qs_1 and followed by inverse transform to provide the reconstruction $\tilde{x}_{\text{rec1}}$. Finally, $\tilde{x}_{\text{rec1}}$ is filtered, and the reference x_{ref1} is used to update the frame buffer. Notice that the Flex SP scheme can output either the reconstructed reference x_{ref1} or the reconstructed high-quality image before *Qs* modules for display.

The decoding of the secondary SP frame is similar to that of the primary SP frame shown in Fig. 4 despite of the modules marked with the dashed block. It means that in case of switching, there is no additional option to reconstruct a high quality image for display.

Please note that in the above encoder and decoder, the quantizer *Qs* and *Qp* can be different while their corresponding de-quantizers are the same [7]. The major difference between these two quantizers is that, quantizer *Qs* should subject to

$$\begin{aligned} Q_S\left(x + Q_S^{-1}(Y)\right) &= Q_S(x) + Q_S\left(Q_S^{-1}(Y)\right) \\ &= Q_S(x) + Y \end{aligned} \tag{1}$$

while quantizer *Qp* does not necessarily have such a property. We will see that this property is important to enable a drift-free switching in the following. Another necessary property of *Qs* and *Qp* de-quantizers is

$$Q^{-1}(Q(x) + Q(y)) = Q^{-1}(Q(x)) + Q^{-1}(Q(y)). \tag{2}$$

Assume that the switching is from S_1 to S_2 at time t, then the reconstructed reference at decoder at time $t-1$ is $x_{\text{ref1}}(t-1)$, and Stream S_{12} at time t is transmitted to decoder. With the proposed SP decoder, the reconstructed reference at time t is

$$\tilde{X}_{\text{rec12}}(t) = Q_{S2}^{-1}\left(X_{\text{srec12}}(t)\right) \tag{3}$$

where

$$\begin{aligned} X_{\text{srec12}}(t) &= Q_{S2}\left(X_{\text{pred12}}(t) + Q_{S2}^{-1}(X_{\text{err12}}(t))\right) \\ &= Q_{S2}\left(X_{\text{pred12}}(t)\right) + Q_{S2}\left(Q_{S2}^{-1}(X_{\text{err12}}(t))\right) \\ &= Q_{S2}(X_{\text{pred12}}(t)) + X_{\text{err12}}(t) \\ &= Q_{S2}(\text{DCT}(\text{MC}(x_{\text{ref1}}(t-1)))) + X_{\text{err12}}(t). \end{aligned} \tag{4}$$

Here, the property (1) of *Qs* is used. As discussed in the encoding process

$$\begin{aligned} X_{\text{srec2}}(t) &= Q_{S2}(X_{\text{pred12}}(t)) + X_{\text{err12}}(t) \\ &= Q_{S2}(\text{DCT}(\text{MC}(x_{\text{ref1}}(t-1)))) + X_{\text{err12}}(t). \end{aligned} \tag{5}$$

Thus, $X_{\text{srec12}}(t) = X_{\text{srec2}}(t)$. Clearly, the Flex SP scheme is drift-free.

There are several significant advantages inherently provided by the Flex SP scheme. First, the Flex SP scheme improves the decoded video quality of the primary SP frame. By removing the *Qs* modules on prediction, the predicted residue error is reduced and the bitstream can be shortened. Moreover, the proposed scheme can output a high-quality display image before the *Qs* modules in the reconstruction loop. This will significantly improve the video quality of the primary SP frame. Meanwhile, the complexity of the primary SP encoder in the Flex SP scheme is lessened by reducing the number of quantization module. As the *Qs* modules can be used in either the primary SP frame encoder or the secondary SP frame encoder, it provides complexity flexibility for different application scenarios.

Moreover, in the Flex SP scheme, different quantization parameters *Qs* can be used in Stream S_1 and Stream S_2. In other words, Stream S_{12} for switching from Stream 1 to Stream 2 is only related to Qs_2. Similarly, Stream S_{21} for switching from Stream S_2 to Stream S_1 is merely related to

Qs_1. By optimizing the parameters Qs_1 and Qs_2 independently, the proposed scheme is able to minimize the size of the switching bitstream while preserving the coding efficiency of S_1 and S_2.

Another desired feature provided by the Flex SP scheme is that the switching points for up-switching and down-switching can be decoupled as well. This means that we can encode more down-switching points than up-switching points to suit the TCP-friendly protocols. Furthermore, the performance of the source bitstream which is switched from is controllable by independently setting the *Qs* in its reconstruction loop. In the case that it is not used as a switching point from other streams, the source bitstream can be coded as P-frame.

III. Hybrid SP Coding Scheme

As discussed in the previous section, if the client is sufficiently powerful, then the Flex SP scheme can significantly improve the decoded video quality with additional IDCT and de-blocking filter and lessen the bit rate of the encoded bitstream by taking advantage of the unquantized prediction. However, in case the reconstructed reference is directly output as display, the Flex SP scheme will not always provide improved quality [12]. Therefore, in this section, a hybrid SP scheme is presented to further improve the quality of both the display image and the reconstructed reference.

The effect of the *Qs* modules on reconstructed reference is first studied here. Normally, for a good tradeoff between coding efficiency of the primary SP bitstream and the size of the switching bitstream, the quantization step of the *Qs* modules is chosen to be close to that of the *Qp* modules in the target bitstream (the bitstream to switch to). Let us assume that the quantization step of *Qs* is equal to that of *Qp* in our analysis.

In the SP coding method, the reconstructed reference in the DCT domain $X_{\rm rec}$ is

$$X_{\rm rec} = Qs^{-1}Qs\left(X_{\rm pred} + Qp^{-1}(X_{\rm perr})\right). \tag{6}$$

In Case 1, the quantized prediction is used. Thus, the residue $X_{\rm perr(1)}$ is

$$X_{\rm perr(1)} = Qp\left(X_{\rm org} - Qs^{-1}Qs(X_{\rm pred})\right) \tag{7}$$

whereas, in Case 2, which is the same as for the Flex SP method, the residue $X_{\rm perr(2)}$ is

$$X_{\rm perr(2)} = Qp(X_{\rm org} + X_{\rm pred}).. \tag{8}$$

Thus, in Case 1, the distortion between the original frame and reconstructed reference in DCT domain is given by

$$\begin{aligned} E_{(1)} &= X_{\rm org} - X_{\rm rec,org} \\ &= X_{\rm org} - Qs^{-1} \\ &\quad \times Qs\left(X_{\rm pred} + Qp^{-1}Qp(X_{\rm org} - Qs^{-1}Qs(X_{\rm pred}))\right) \end{aligned} \tag{9}$$

and the corresponding distortion in Case 2 is given by

$$\begin{aligned} E_{(2)} &= X_{\rm org} - X_{\rm rec,Flex} \\ &= X_{\rm org} - Qs^{-1}Qs \\ &\quad \times \left(X_{\rm pred} + Qp^{-1}Qp(X_{\rm org} - X_{\rm pred})\right). \end{aligned} \tag{10}$$

For the typical case where the quantization steps of *Qs* and *Qp* are the same, (9) becomes (considering the properties of the *Qs* quantizer and de-quantizer)

$$\begin{aligned} E_{(1)} &= X_{\rm org} - Qs^{-1}Qs(X_{\rm pred}) \\ &\quad - Qp^{-1}Qp\left(X_{\rm org} - Qs^{-1}Qs(X_{\rm pred})\right) \\ &= Z - Qp^{-1}Qp(Z) \end{aligned} \tag{11}$$

and (10) becomes

$$\begin{aligned} E_{(2)} &= X_{\rm org} - Qs^{-1}Qs(X_{\rm pred}) \\ &\quad - Qp^{-1}Qp\left(X_{\rm org} - Qs^{-1}Qs(X_{\rm pred}) - E_{Qs}\right) \\ &= Z - Qp^{-1}Qp(Z - E_{Qs}) \end{aligned} \tag{12}$$

where $Z = X_{\rm org} - Qs^{-1}Qs(X_{\rm pred})$ and $E_{Qs} = X_{\rm pred} - Qs^{-1}Qs(X_{\rm pred})$.

Clearly, the distortion in Case 1 is the quantization error of Z, while the distortion in Case 2 is the error between Z and its biased quantization. Commonly, Case 1 has less distortion, which indicates that using the quantized prediction can indeed benefit the quantized reconstruction. However, the Flex SP scheme does produce smaller predicted residue as well as a shorter bitstream. Therefore, when the reconstructed reference is output as display, the overall coding efficiency of the Flex SP scheme may sometimes decrease a little, while in other cases it may slightly increase.

Accordingly, the coding efficiency of reconstruction as well as display can be improved if the prediction, either quantized or unquantized, is selected properly. Thus, the Hybrid SP coding scheme is proposed here to improve the coding efficiency by optimizing the temporal prediction. Fig. 5 depicts the block diagram of the primary SP frame encoder in the Hybrid SP scheme. There is a switch in the primary SP encoder denoted by a dashed block, which indicates that either the temporal prediction $X_{\rm pred}$ or the de-quantized version $\tilde{X}_{\rm pred}$ is selected to form the prediction for motion compensation. Notice that, no matter which temporal prediction is chosen to generate the predicted residue, the prediction $X_{\rm pred}$ is always used in $X_{\rm rec}$ reconstruction. This provides two notable advantages: 1) no overhead bits need to be transmitted for informing the decoder which temporal prediction is chosen and 2) the motion estimation is optimized coefficient by coefficient.

Two encoding modes are defined for primary SP frame (as shown in Fig. 6), namely, the Q-mode and the N-mode, corresponding to two different prediction methods. In Fig. 6, rectangular boxes filled with horizontal bars denote the DCT coefficient of prediction $X_{\rm pred}$. Rectangular boxes with vertical bars denote the DCT coefficient of de-quantized prediction $\tilde{X}_{\rm pred}$. Rectangular boxes with slanted bars denote the reconstructed DCT coefficient. Solid arrows with solid lines are for the motion estimation and hollow arrows with solid lines are for the reconstruction. When the N-mode is chosen, the motion estimation uses the prediction $X_{\rm pred}$; otherwise, the de-quantized

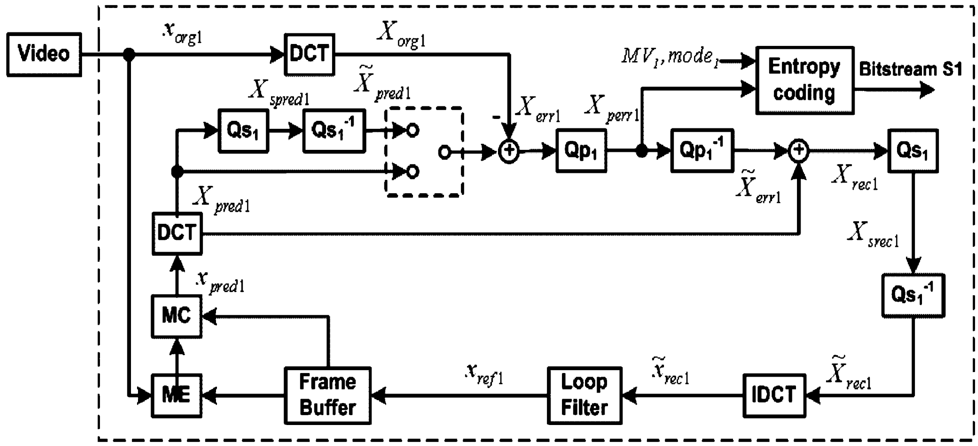


Fig. 5. Encoder of the Hybrid SP scheme.

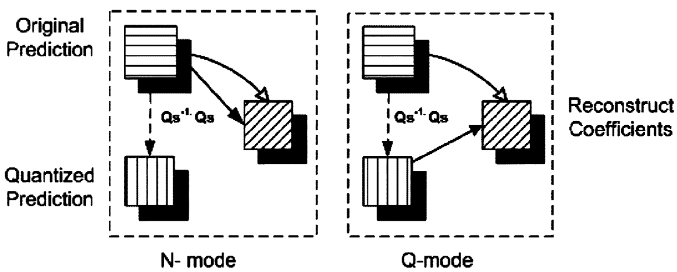


Fig. 6. Two prediction modes in the Hybrid SP scheme.

prediction $\tilde{X}_{\text{pred}}$ is used in the motion estimation. In both cases, X_{pred} is used in the reconstruction of the DCT coefficients.

An R-D optimization algorithm is proposed to select the prediction for each DCT coefficient. The ideal method should take into account both the reconstruction reference and the display image in the R-D model, which results in a multiobjective optimization problem. To avoid intensive computation, the proposed R-D optimization algorithm only tries to optimize the quality of the display image. As the display image has a strong correlation with the reconstructed reference, the proposed R-D algorithm can improve the quality of the reconstructed reference as well.

The Lagrangian formulation of the R-D optimization problem is given as follows:

$$\text{argmin}_{X_p \epsilon \{X_{\text{pred}}, \tilde{X}_{\text{pred}}\}} D_{X_p} + \lambda \cdot R_{X_p}. \tag{13}$$

Here, X_p is the optimal prediction which is either X_{pred} or $\tilde{X}_{\text{pred}} \cdot \lambda$ is the Lagrange multiplier related to the quantization step *Qp*. Thomas *et al.* gives an empirical formula between λ and *Qp* in [14]. D_{X_p} is the distortion of the display image. The sum of absolute difference (SAD) metric is used here, i.e.,

$$D_{X_p} = \text{SAD}\left(X_{\text{org}} - Qp^{-1}(L(X_p)) - X_{\text{pred}}\right) \tag{14}$$

where $Qp^{-1}()$ is the de-quantization process and $L(X_p)$ is the quantized prediction residue of each DCT coefficient, which is defined as

$$L(X_p) = Qp(X_{\text{org}} - X_p). \tag{15}$$

Here, R_{X_P} is the number of bits spent on $L(X_p)$, which can be obtained by looking up the VLC table. If the predicted residue is coded by the run-length coding method, then the actual number of bits spent on each residue should be closely related to its context. In particular, when $L(X_P)$ is zero, the corresponding value of R_{X_p} is determined by the increasing number of successive zeros. On the other hand, when CABAC is used, R_{X_p} has to be calculated by a precoded process. To simplify the computation, R_{X_p} in the proposed scheme is estimated coefficient by coefficient considering little of its context, i.e., the number of bits for coding $L(X_P)$ is estimated without the context model of CABAC. Particularly, if $L(X_P)$ is zero, R_{X_p} is forced to be zero.

Except for the two temporal prediction modes used in the motion estimation, the Hybrid SP scheme is the same as the Flex SP scheme. Therefore, it maintains all of the advantages that we discussed in the previous section. Through the R-D optimization, the Hybrid SP scheme further improves the quality of the display image and the reconstructed reference. The proposal of the Hybrid SP scheme has been accepted by the JVT standard [15], [20]. As the decoding process of the Hybrid SP scheme is similar to that of the Flex SP scheme, then the decoder shown in Fig. 4 can decode the bitstream generated by either of them. Both schemes discussed in this paper comply with the JVT standard.

IV. Comparisons Among the SP Coding Schemes

Some experiments have been performed to verify the performance of the proposed two SP schemes. The coding efficiencies of the proposed two SP coding schemes with respect to a common SP scheme shown in Fig. 2 are evaluated. The JM61c software is used in the experiments [16]. The sequences Foreman, Container, and Coastguard in QCIF format are coded at 10 Hz. The Mobile and Tempete sequence in CIF format are coded at 30 Hz. In the experiments, only the first frame is coded as I-frame, and other frames are coded as SP frames. The parameters at the encoder are set as follows.

- RD optimization: Enable.
- Hadamard transform: Enable.
- Search Range: 16.
- MC: $1/4$ pixel.
- Reference number: 1.
- B frame: No.
- Inter and Intra mode: All.
- Entropy coding: CAVLC.

TABLE I
COMPARISONS OF BOTH THE RECONSTRUCTED REFERENCES AND THE DISPLAY OF THE COMMON SP SCHEME AND THE TWO PROPOSED SP METHODS

Seq.	QP	Common SP		Flex SP			Hybird SP			ave. saving			
		PSNRY	bitrate	PSNRY (Ref)	PSNRY (Dis)	bitrate	PSNRY (Ref)	PSNRY (Dis)	bitrate	Flex SP (Ref)	Hybird SP (Ref)	Flex SP (Dis)	Hybird SP (Dis)
Fore man (QCIF)	24	37.694	155700	36.905	37.903	146275	36.756	37.478	135822				
	28	34.634	98243	34.007	34.892	91694	33.872	34.546	85358	-0.31%	4.00%	10.56%	12.78%
	32	31.471	58814	30.989	31.718	54218	30.862	31.453	50711				
	36	28.441	33881	28.128	28.668	31390	27.986	28.448	29441				
	40	25.546	18198	25.302	25.601	17390	25.266	25.525	16267				
Coast guard (QCIF)	24	36.702	254527	35.858	36.775	249823	35.638	36.151	227562				
	28	33.469	147627	32.750	33.508	141319	32.547	33.012	126422	-6.60%	1.07%	7.40%	10.58%
	32	30.126	70218	29.546	30.126	65171	29.450	29.856	59199				
	36	27.202	31890	26.823	27.257	29488	26.780	27.108	26659				
	40	24.733	14101	24.432	24.694	12927	24.444	24.652	12048				
Conta iner (QCIF)	24	38.022	60790	37.612	38.031	58307	37.479	37.765	53576				
	28	35.184	33564	34.819	35.228	32204	34.740	35.049	29884	-1.80%	3.35%	4.63%	8.52%
	32	32.007	19100	31.675	32.051	18230	31.611	31.931	16990				
	36	28.742	9629	28.471	28.745	9279	28.406	28.649	8749				
	40	26.029	4887	25.822	25.969	4583	25.797	25.919	4375				
Mobil e (CIF)	24	36.523	392752	35.230	36.645	3758637	35.048	35.833	3414556				
	28	32.825	248057	31.718	33.031	2327750	31.600	32.451	2101679	-4.06%	4.45%	12.28%	16.59%
	32	28.475	132247	27.720	28.798	1225581	27.755	28.591	1133618				
	36	24.565	655838	24.091	24.909	603404	24.182	24.889	563556				
	40	21.447	335894	21.112	21.619	304311	21.139	21.595	282470				
Temp ete (CIF)	24	37.252	292514	36.168	37.375	2819905	35.980	36.657	2548058				
	28	33.797	175401	32.866	33.930	1651976	32.718	33.404	1478161	-5.93%	2.16%	9.16%	13.00%
	32	29.893	880926	29.226	30.042	818021	29.185	29.805	747714				
	36	26.489	406465	26.031	26.578	374879	26.014	26.464	344850				
	40	23.612	173693	23.325	23.551	159188	23.282	23.467	147979				
Ave.										**-3.74%**	**3.01%**	**8.81%**	**12.29%**

These parameters were specified by the SP core experiment in the JVT standard [11]. Since the JVT software does not provide a rate-control method, a fixed quantization parameter is used in the coding process. This brings some trouble when comparing different techniques. JVT recommends using the Bjontegaard measurement method [17], in which four pairs of rate and PSNR at different quantization parameters are used to calculate the average bit savings and the average PSNR gain. The same method is adopted in the experiments to evaluate the three SP schemes by the average bit savings. The four quantization parameters are 28, 32, 36, and 40.

When *Qs* is set equal to *Qp*, the reconstructed references between the common SP scheme and the two proposed SP schemes are compared, respectively, in Table I (denoted by the columns marked with Ref). In this case, the reconstruct reference is used as the display image. Thus, the two proposed SP schemes do not increase any decoding complexity. Despite the advantage of flexibility of switching, there is a slight loss of the coding efficiency of the Flex SP scheme. In other words, with regard to the same reconstruct quality, the average bit of the Flex SP scheme increases by about 3.74%. However, such a shortage is overcome by the Hybrid SP method, which can save up to 4.45% bits for Mobile CIF at different quantization parameters. For all test sequences, the average bit saving of the Hybrid SP scheme are 3.01% compared with the common SP method when the same quality is provided.

If the additional inverse DCT and post filter are allowed at the decoder, the two proposed SP schemes can output a high-quality image for display. The reconstructed reference of the common SP scheme and the reconstructed display of the two proposed SP schemes are compared in Table I (denoted by the columns marked with Dis), where *Qs* is set equal to *Qp*. In the case of the same display quality, the average bit savings for all testing sequences are 8.81% and 12.29% of the Flex SP scheme and the Hybrid SP scheme, respectively. For Mobile CIF, Tempete CIF, and Foreman QCIF, the average bit savings of the proposed two SP methods is more than 10%. In particular, a 16.59% saving in bits is achieved by the Hybrid SP method for Mobile CIF.

Fig. 7 shows the average number of bits used to code the secondary SP frames. Three sequences with the QCIF format are coded, and eight switching scenarios given in Table II are simulated in this experiment. As indicated by Fig. 7, the sizes of the switching bitstream of the three compared SP methods are at the same level. To further test the coding performance of the Hybrid SP scheme, the *Qs* sets from $Qp-4$ to $Qp+3$ are evaluated by reconstructed reference. As shown in Table III, the coding performance of the Hybrid SP scheme is consistently better than that of the common SP scheme.

V. DISCUSSION ON QUANTIZATION PARAMETERS FOR SWITCHING

How to choose the quantization parameters for switching is an interesting topic in all SP coding schemes. The quantization parameter *Qp* is used to control the bit rate of primary SP frames, while the use of different values of *Qs* allows us to trade off between the coding performance of primary SP frames and the bits of secondary SP frames. If the difference between the reconstruction of a source bitstream and the temporal prediction from the destination bitstream was directly coded losslessly, the size of the switching bitstream would be quite huge and even unacceptable in many applications. This is why the *Qs* modules are introduced into the primary SP frame coding. Commonly, the larger the *Qs* is, the smaller the switching bitstream is, i.e., the bit rate of the generated switching bitstream can be sufficiently lowered by selecting large *Qs*. However, a large *Qs* will lead to a low coding efficiency of the primary SP frames. Normally, a small *Qs* is preferred to maintain the video quality of the primary SP frame, as the secondary SP frame is only sent in

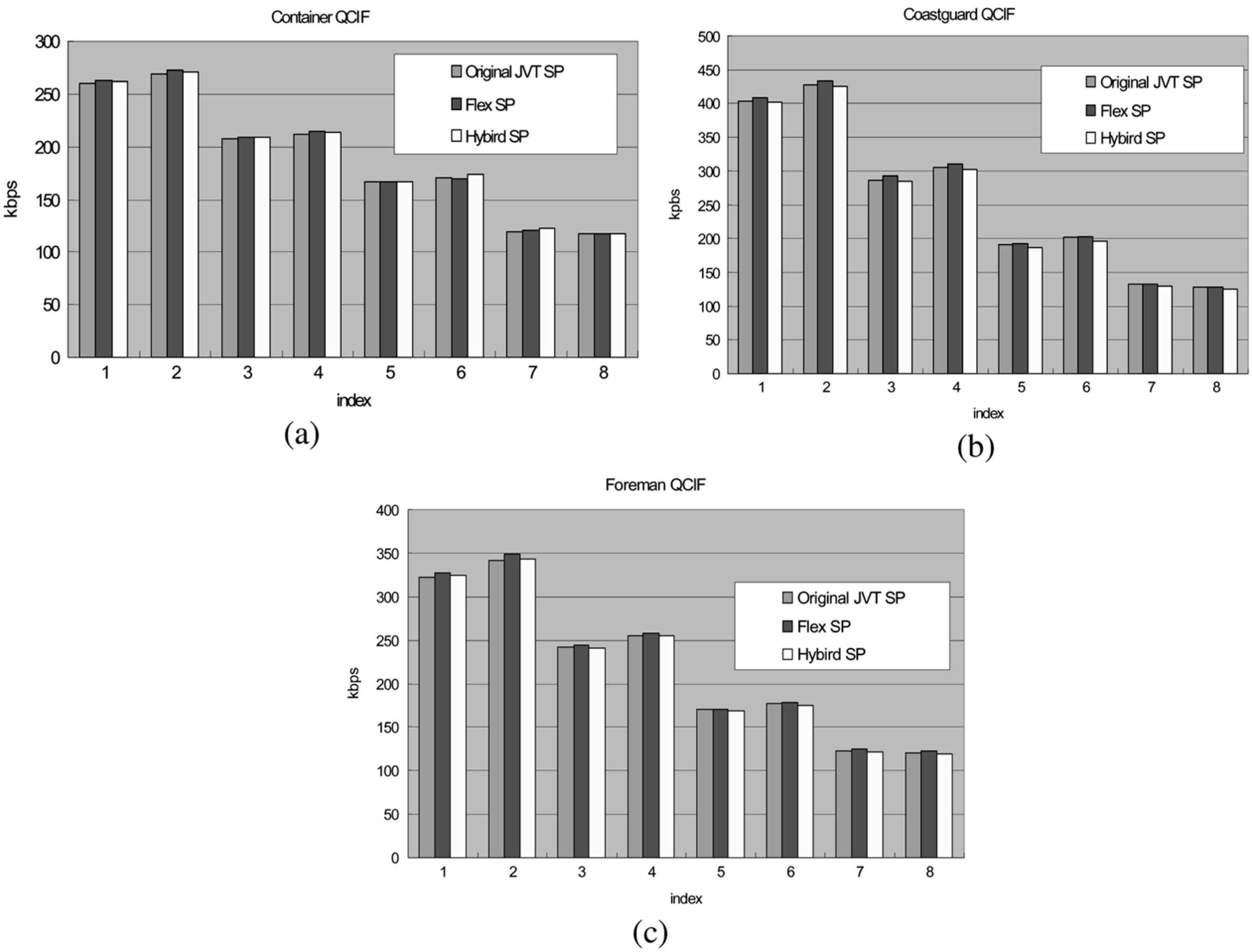


Fig. 7. Comparisons of the average bit rate of the secondary SP frame. Here, the index 1 to 8 represents the switching scenario shown in Table II. (a) Container. (b) Coastguard. (c) Foreman.

TABLE II
SWITCHING SCENARIOS CORRESPONDING TO FIG. 7

index	1	2	3	4	5	6	7	8
Source (Qp/Qs)	24/ 24	28/ 24	28/ 28	32/ 28	32/ 32	36/ 32	40/ 36	36/ 36
Destination (Qp/Qs)	28/ 24	24/ 24	32/ 28	28/ 28	36/ 32	32/ 32	36/ 36	40/ 36

the case of switching. In the following subsections, the effect of *Qs* is evaluated both on the coding performance of primary SP frames and the bit rate of secondary SP frames.

A. Coding Performance of Primary SP Frame

Two test sequences, Foreman and Coastguard at QCIF format, are used to test the performance. Four different rules, as described in Table IV, are used to evaluate the impact of *Qs* on the coding performance of primary SP frames. The quantization parameter *Qp* is set to 24, 28, 32, 36, 40 respectively. The encoding frame rate is 10 Hz.

In the first experiment, only the first frame is coded as an I-frame, while other frames are all coded as SP frames. Moreover, the test sequences are also coded in pure P-frames (except for the first I-frame) to provide anchors for the evaluation. Fig. 8(a) shows the R-D curves of the four schemes together with the anchor. It can be seen that a smaller *Qs* results in a higher performance of primary SP frames while larger *Qs* results in a lower coding efficiency. This is consistent with the aforementioned statement. Notice that Scheme III provides much better performance of primary SP frames along with the decrease on *Qs* compared with Scheme I and Scheme II. However, setting *Qs* to ever lower, as shown in Scheme IV, the improvement in coding efficiency is very limited but will result in a much heavier switching bitstream.

To further test the *Qs* effect on the coding efficiency, we present another experiment when SP frames are inserted periodically at intervals of ten frames. Other settings are the same as that in the above experiment. This scenario is more applicable for bitstream switching and random access in real applications. Fig. 8(b) shows the experimental results. It can be seen that the curve of Scheme IV is very close to that of the anchor. Further reducing the Qs to $Qp-12$ can gain a neglible amount on primary SP-frame coding. A similar result was also illustrated in [18].

B. Coding Performance of Secondary SP Frame

Two test sequences, Foreman and Coastguard at QCIF format, are utilized in this experiment. Only the first frame is

TABLE III
COMPARISONS OF THE RECONSTRUCTED REFERENCE OF THE COMMON SP SCHEME AND THE HYBRID SP SCHEMES WITH *QS* SET FROM $QP-4$ TO $QP+3$. THE TEST SEQUENCE IS FOREMAN (QCIF).

QPSP	QP	Common SP		Hybird SP		ave. bits saving	ave. dB gain
		PSNRY	bitrate	PSNRY	bitrate		
QP-1	24	*37.361*	*151554*	36.850	132766		
	28	*34.469*	*95218*	33.989	83278	6.45%	0.285
	32	*31.337*	*56355*	30.977	49562		
	36	*28.340*	*32294*	28.097	28476		
	40	*25.424*	*17548*	25.219	15856		
QP-2	24	*37.429*	*144970*	37.006	129704		
	28	*34.476*	*91112*	34.090	81283	5.06%	0.242
	32	*31.386*	*53536*	31.071	48109		
	36	*28.432*	*30776*	28.189	27904		
	40	*25.512*	*17366*	25.345	15786		
QP-3	24	*37.585*	*139241*	37.157	126842		
	28	*34.651*	*86287*	34.275	79309	4.16%	0.187
	32	*31.474*	*51603*	31.156	46955		
	36	*28.528*	*30159*	28.265	27395		
	40	*25.602*	*16842*	25.445	15264		
QP-4	24	*37.717*	*135358*	37.301	125505		
	28	*34.757*	*84312*	34.357	77577	2.71%	0.122
	32	*31.605*	*49458*	31.302	45591		
	36	*28.629*	*28988*	28.419	27187		
	40	*25.652*	*16514*	25.521	15389		
QP+1	24	*37.663*	*162625*	36.636	138950		
	28	*34.618*	*105236*	33.734	88918	4.42%	0.201
	32	*31.439*	*61650*	30.772	52418		
	36	*28.440*	*34798*	27.970	30025		
	40	*25.495*	*19062*	25.188	16693		
QP+2	24	*37.505*	*173664*	36.417	144109		
	28	*34.512*	*111230*	33.618	92838	5.93%	0.271
	32	*31.397*	*65624*	30.677	54658		
	36	*28.376*	*37410*	27.880	31255		
	40	*25.393*	*20034*	25.084	17304		
QP+3	24	*37.293*	*183606*	36.247	150599		
	28	*34.384*	*118898*	33.480	97046	8.14%	0.397
	32	*31.245*	*72257*	30.540	58307		
	36	*28.278*	*40184*	27.768	33170		
	40	*25.354*	*21186*	25.054	17665		
total						5.20%	0.26

TABLE IV
QS FOR EACH SCHEME

Scheme	I	II	III	IV
Qs Setting	Qp	Qp-2	Qp-6	Qp-12

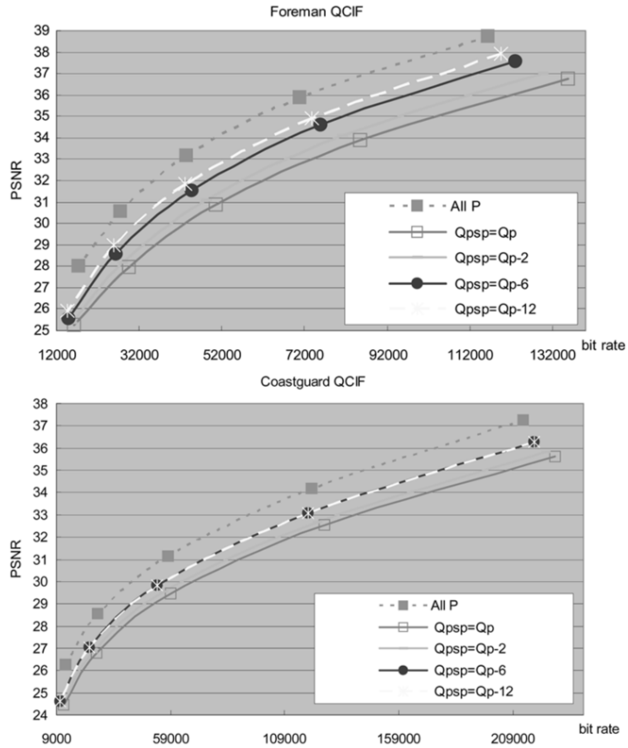
coded as I-frame, and the others are coded as SP frame without an intra macroblock.

The switching-down scenario is tested first. The *Qp* parameters of source video coding and destination video coding are set to be 0 and 32, respectively. The *Qs* parameters for the SP-frame coding are from 24 to 40 at intervals of 2. Experimental results are shown in Fig. 9. The curves marked NormalSP are the R-D curves of the primary SP frame (*Qp* equals 32) with different *Qs* settings. The curves marked SwitchSP (α) represent the bit rates of the secondary SP frames with a varying weighted factor α. Here, the weight factor α is the probability of the switching-up. For example, SwitchSP(0.1) simulates the applications in which 10% secondary SP frames may be used. Assuming that the total bit rate of the secondary SP bitstream $R(1)$ is C, then the bit rate of SwitchSP (α) can be calculated by

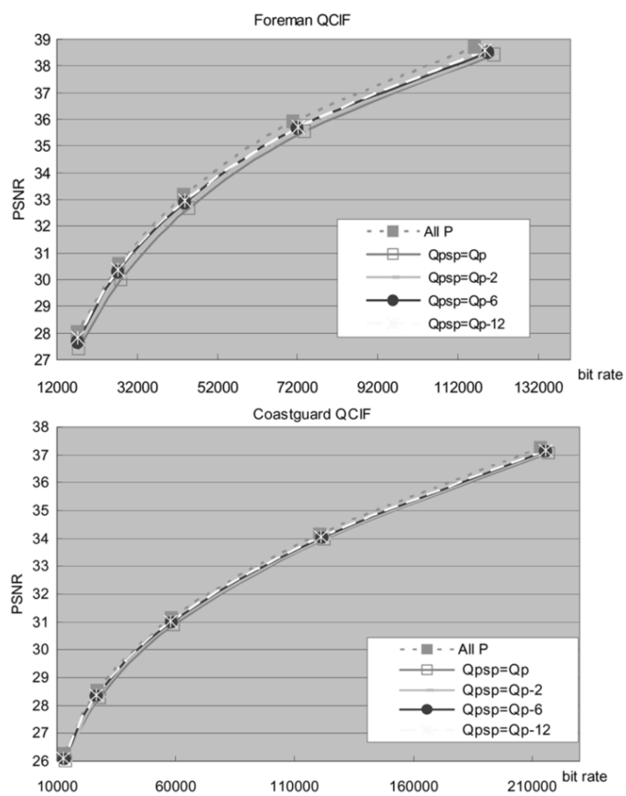
$$R(\alpha) = \alpha \times R(1) = \alpha \times C.$$

The top diagram of Fig. 9 shows the experimental results for the Foreman sequence. We can observe that there are cross points between the R-D curves of the primary and secondary SP bitstreams, which indicates the balance points of coding performances of primary and secondary SP frames. Accordingly, when the weight factor α is 0.1, a smaller *Qs* equal to 24 is preferred. On the other hand, when the weight factor increases to 0.5, it is more efficient to set *Qs* to 36. Consistent results are also obtained in the case of the Coastguard sequence, as illustrated in the bottom of Fig. 9.

In case of switching up, the *Qs* parameters of source video coding and destination video coding are set to 40 and 32, respectively. The *Qs* parameters for SP frames coding are also from 24 to 40 at intervals of 2. The results are shown in Fig. 10. As the



(a)



(b)

Fig. 8. Coding efficiency of primary a SP frame with different Qs settings. (a) All frames are coded as SP frames (except for the first frame) (b) SP frames are coded at intervals of ten frames.

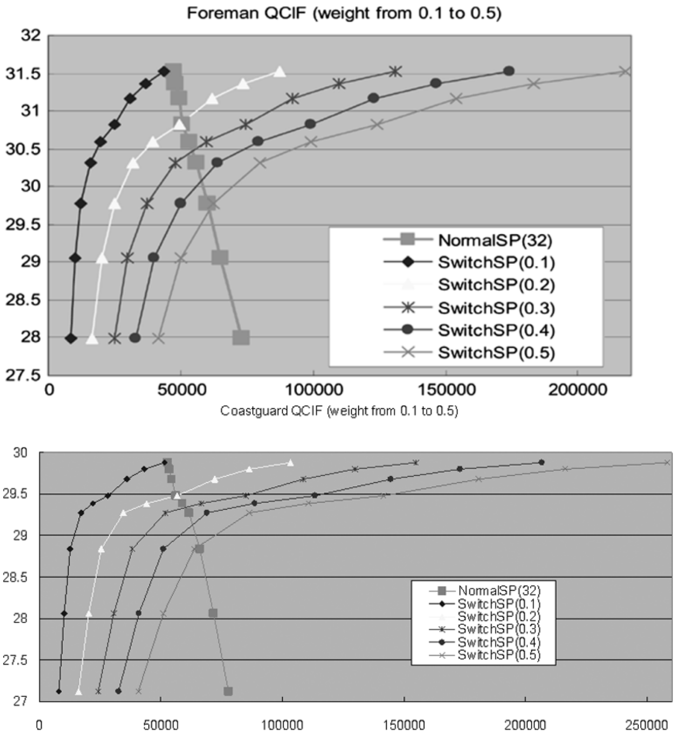


Fig. 9. R-D of the primary SP frame and the secondary SP frame when switching down.

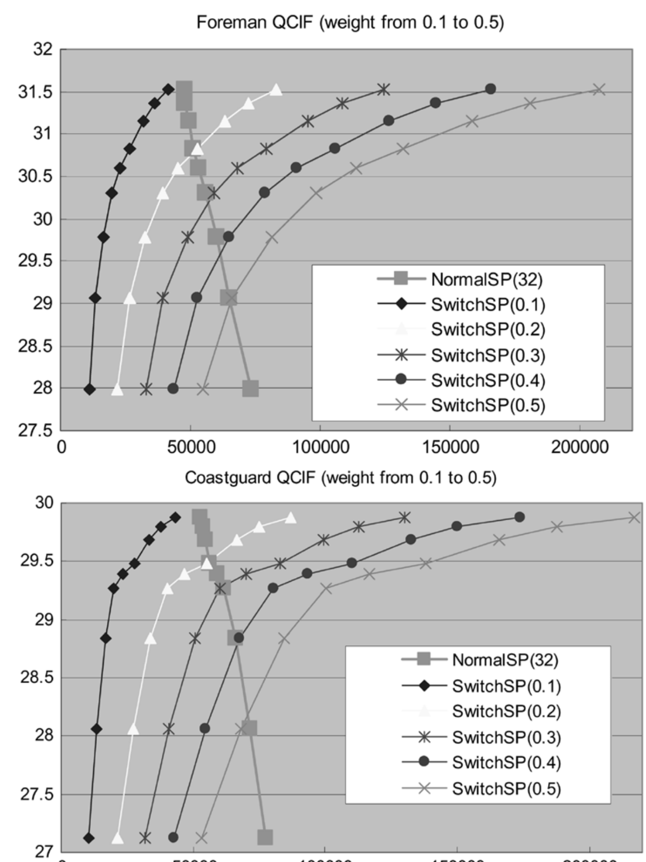


Fig. 10. R-D of primary and secondary SP frames in switching up.

probability of switching-up rises from 0.1 to 0.5, the suitable Qs parameter increases from 24 to 36.

Deriving from the above analyses and experimental verifications, Qs selection is related to both Qs's of two primary SP bitstreams and the possible ratio of switching. However, we can see that, regardless of whether it is switching up or switching down, after these parameters are decided by the applications, we can use the same empirical method to choose Qs to achieve a better tradeoff between good coding performance of primary SP frames and small size of switching bits.

VI. Conclusion

Two SP coding schemes are proposed in this paper to efficiently compress video contents into bitstreams that support drift-free switching at predictive frames. In the Flex SP scheme, the quantization module on temporal prediction is removed, and a high-quality image can be outputted for display before the quantization Qs. Thus, the Flex SP scheme significantly saves up to 12% bits at the same display quality. Furthermore, it decouples the quantization parameters for up-switching and down-switching and is able to insert more down-switching points than up-switching points. Meanwhile, the size of the down-switching bitstream can be much smaller than that of the up-switching one by adjusting the quantization parameter. These features are very desirable for the TCP-friendly protocols currently used in most streaming systems.

Moreover, the Hybrid SP scheme is presented by taking advantage of the optimal selection of temporal prediction. It improves the quality of the display image as well as the reconstructed reference by two DCT predictive coding modes. No overhead bits need to be transmitted to inform the decoder of the coding mode. The corresponding R-D algorithm is also proposed to optimally select the coding mode for each DCT coefficient. In contract to the nonoptimal SP scheme, the Hybrid SP scheme can significantly save up to 16% bits at the same quality. The average bit savings for all testing sequences are 12.29%. Even in the case of the same complexity, i.e., a reconstructed reference used for display, the Hybrid SP scheme can still save 3.01% bits on average by R-D optimization. In addition, the desirable features provided in the Flex SP scheme are still readily supported by the Hybrid SP method. The generated bitstreams by both proposed schemes can be decoded successfully with the decoder that complies with MPEG-4 AVC/H.264.

In this paper, we also investigated how to choose the quantization parameters for switching. The experimental results show that it is tightly related to Qp of primary SP bitstreams and switching frequency. An empirical method is proposed to choose Qs to achieve a better tradeoff between good coding efficiency of primary SP frames and small size of switching bits.

Acknowledgment

The authors would like to thank G. Sullivan for many valuable discussions and suggestions in proposing the two SP schemes to the JVT standard. The authors would also like to thank R. Kurceren for making available the software of their SP scheme and cross-checking the experimental results. The authors are also grateful to the editor and anonymous reviewers for their time and constructive comments.

References

[1] J. Lu, "Signal processing for internet video streaming: A review," in *Proc. SPIE Image Video Commun. Process.*, Australia, 2000, vol. 3974, pp. 246–258.

[2] G. Conklin, G. Greenbaum, K. Lillevold, A. Lippman, and Y. Reznik, "Video coding for streaming media delivery on the internet," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 3, pp. 269–281, Mar. 2001.

[3] D. Wu, Y. Hou, W. Zhu, Y. Zhang, and J. Peha, "Streaming video over the internet: Approaches and directions," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 3, pp. 282–300, Mar. 2001.

[4] B. Girod, N. Farber, and U. Horn, "Scalable codec architectures for internet video on demand," in *Proc. Asilomar Conf. Signals Syst.*, 1997, pp. 357–361.

[5] N. Farber and B. Girod, "Robust H.263 compatible video transmission for mobile access to video servers," in *Proc. Int. Conf. Image Process.*, Oct. 1997, vol. 2, pp. 73–76.

[6] M. Jarczewicz and R. Kurceren, A Proposal for SP-Frames Nokie Research Center, ITU-T Q.6/SG 16, VCEG-L27, 2001.

[7] R. Kurceren and M. Karczewisz, Improved SP-Frame Encoding Austin, TX, ITU-T Q.6/SG 16, VCEG-M73, Apr. 2001.

[8] X. Sun, S. Li, F. Wu, G. Shen, and W. Gao, "Efficient and flexible drift-free video bitstream switching at predictive frames," in *Proc. Int. Conf. Multimedia Expo (ICME)*, Lausanne, Switzerland, 2002, pp. 541–544.

[9] "H.26L Test model long-term number 9 (TML-9) draft0," in *ITU-T Video Coding Experts Group 14th Meeting*, Santa Barbara, CA, Sep. 2001, Doc. EG-N83 d1,.

[10] X. Sun, F. Wu, S. Li, W. Gao, and Y.-Q. Zhang, "Improved SP coding technique," in *Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG* (in document JVT-B097), Geneva, Switzerland, Jan. 2002.

[11] F. Wu, S. Li, R. Kurceren, and J. Lainema, "The core experiment on the SP frame coding," in *Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG* (in document JVT-B112), Geneva, Switzerland, Jan. 2002.

[12] X. Sun, F. Wu, and R. Kurceren, "Core experiment results on SP," in *Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG* (in document JVT-C126), Fairfax, VA, May 2002.

[13] X. Sun, F. Wu, S. Li, and R. Kurceren, "The improved JVT-B097 SP coding scheme," in *Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG* (in document JVT-C114), Fairfax, VA, May 2002.

[14] T. Wiegand and B. Girod, "Lagrange multiplier selection in hybrid video coder control," in *Proc. Int. Conf. Image Process.*, Thessaloniki, Greece, Oct. 2001, pp. 542–545.

[15] , "Joint final committee draft (JFCD) of joint video specification," in *Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG* (in JVT, document JVT-D157), Klagenfurt, Germany, Jul. 2002.

[16] , [Online]. Available: http://standard.pictel.com/ftp/video-site/h26L

[17] G. Bjontegaard, "Calculation of average PSNR differences between RD curves," in *ITU-T Video Coding Experts Group (VCEG) Meeting* (in document VCMG-33), Austin, TX, Apr. 2–4, 2001.

[18] M. Karczewicz and R. Kurceren, "The SP- and SI-frames design for H.264/AVC," *IEEE Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 637–644, Jul. 2003.

[19] R. Kurceren and M. Karczewicz, "Further results for SP-frames," in *ITU-T Video Coding Experts Group Meeting* (in Doc. VCEG-M-38), Austin, TX, Apr. 02–04, 2001.

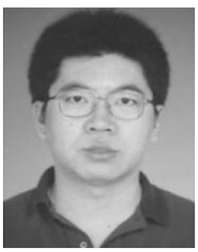
[20] G. Sullivan, A. Luthra, and T. Wiegand, "(Draft) report of fairfax meeting," in *Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG* (in document JVT-C001d3), Fairfax, VA, May 2002.

Xiaoyan Sun received the B.S., M.S., and Ph.D. degrees in computer science from the Harbin Institute of Technology, Harbin, China, in 1997, 1999, and 2004, respectively.

She has been an Associate Researcher with Microsoft Research Asia, Beijing, China, since 2004. She has published approximately 20 conference and journal papers and submitted several proposals to MPEG-4 and H.264. Her research interests include video/image coding, video streaming, and multimedia processing.

Feng Wu received the B.S. degree in electrical engineering from University of Xi'an Electrical Science and Technology, Xi'an, China, in 1992, and the M.S. and Ph.D. degrees in computer science from the Harbin Institute of Technology, Harbin, China, in 1996 and in 1999, respectively.

He joined Microsoft Research Asia, Beijing, China, as an Associate Researcher in 1999 and was promoted to Researcher in 2001. He has played a major role in the Internet Media Group to develop scalable video coding and streaming technologies. He has authored and coauthored over 60 papers in video compression and contributed some technologies to MPEG-4 and H.264. His research interests include video and audio compression, multimedia transmission, and video segmentation.



Shipeng Li received the B.S. and M.S. degrees from the University of Science and Technology of China (USTC), Hefei, China, in 1988 and 1991, respectively, and the Ph.D. degree from Lehigh University, Bethlehem, PA, in 1996, all in electrical engineering.

He was with the Electrical Engineering Department, USTC, during 1991–1992. He was a Member of Technical Staff with Sarnoff Corporation, Princeton, NJ, during 1996–1999. He has been a Researcher with Microsoft Research Asia, Beijing, China, since May 1999 and has contributed some technologies in MPEG-4 and H.264. His research interests include image/video compression and communications, digital television, multimedia, and wireless communication.

Guobin Shen (S'98–M'01) received the B.S. degree from the Harbin University of Engineering, Harbin, China, in 1994, the M.S. degree from Southeast University, Nanjing, China, in 1997, and the Ph.D. degree from Hong Kong University of Science and Technology (HKUST), Hong Kong, in 2001, all in electrical and electronic engineering.

He was a Research Assistant with HKUST from 1997 to 2001. Since then, he has been with Microsoft Research Asia, Beijing, China, where he is now a Researcher and Project Leader with the Wireless and Networking Group. His research interests include digital image and video signal processing, video coding and streaming, distributed/parallel computing and peer-to-peer networking, general computing on GPU, wireless networking and mobile computing, and media management. He has published approximately 12 journal papers and more than 30 conference papers. He has been granted two U.S. patents and has filed more than a dozen patent applications. He is now serving as TPC member for several international conferences and as reviewer for several journals and many conferences.

Dr. Shen is a member of the Association for Computing Machinery.

Wen Gao received the Ph.D. degree in computer science from the Harbin Institute of Technology, Harbin, China, in 1988, and the Ph.D. degree in electronics engineering from the University of Tokyo, Tokyo, Japan, in 1991.

He joined the faculty of the Harbin Institute of Technology in 1985 and served as Lecturer, Professor, and Chairman of the Department of Computer Science. He joined the Institute of Computing Technology (ICT), Chinese Academy of Sciences, Beijing, as a Professor in 1996. He was the Managing Director of the ICT from 1998 to 1999. From 2000 to 2004, he was appointed as a Professor and Vice President of the Graduate School of the Chinese Academy of Sciences as well as Vice President of the University of Science and Technology of China, Hefei, China. He was a Visiting Scientist with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, in 1993 and a Visiting Scientist with the Artifical Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, in 1995. He was appointed an Honorary Professor with the Department of Computer Science, City University of Hong Kong, Hong Kong, from 1995 to 1997, an Adjunct Professor with the University of Science and Technology of Hong Kong in 2004, and an Honorary Professor with the School of Computing and Engineering, University of Missouri-Kansas City, in 2005. He has served on the editorial boards of several journals and was the Editor-in-Chief of the *Journal of Computers*. He has been involved in many national research and development activities since 1992. He served as the chairman of steering committee for intelligent computing system in national 863 Hi-Tech Programme from 1996 to 2001. He is the head of the Chinese delegation to MPEG and is the chair of the AVS Working Group, which is an entity to make and evaluate the national standard for audio/video coding systems. He has published four books and over 300 technical articles in refereed journals and proceedings in the areas of multimedia, data compression, face recognition, sign language recognition and synthesis, image retrieval, multimodal interface, and bioinformatics.

Dr. Gao was the recipient of national awards for science and technology achievement in 2000, 2002, and 2003.