A Robust Fine Granularity Scalability Using Trellis-Based Predictive Leak

Hsiang-Chun Huang, Chung-Neng Wang, and Tihao Chiang, *Senior Member, IEEE*

***Abstract*—Recently, the MPEG-4 committee has approved the MPEG-4 fine granularity scalability (FGS) profile as a streaming video tool. In this paper, we propose novel techniques to further improve the temporal prediction at the enhancement layer so that coding efficiency is superior to the existing FGS. Our approach utilizes two parameters, the number of bitplanes β ($0 \leq \beta \leq$ maximal number of bitplanes) and the amount of predictive leak α ($0 \leq \alpha \leq 1$), to control the construction of the reference frame at the enhancement layer. These parameters α and β can be selected for each frame to provide tradeoffs between coding efficiency and error drift. Our approach offers a general and flexible framework that allows further optimization. It also includes several well-known motion-compensated FGS techniques as special cases with particular sets of α and β. We analyze the theoretical advantages when parameters α and β are used, and provide an adaptive technique to select α and β, which yields an improved performance as compared to that of fixed parameters. An identical technique is applied to the base layer for further improvement. Our experimental results show over 4 dB improvements in coding efficiency using the MPEG-4 testing conditions. The removal of error propagation is demonstrated with several typical channel transmission scenarios.**

***Index Terms*—Error robustness, fine granularity scalability (FGS), leaky prediction, MPEG-4 video coding, video streaming.**

I. Introduction

RECENTLY, the delivery of multimedia information to mobile device over wireless channels and/or Internet is a challenging problem because multimedia transportation suffers from bandwidth fluctuation, random errors, burst errors, and packet losses [2]. Thus, the MPEG-4 committee has adopted various techniques to address the issue of error-resilient delivery of video information for multimedia communications. However, it is even more challenging to simultaneously stream or multicast video over Internet or wireless channels to a wide variety of devices where it is impossible to optimize video quality for a particular device, bit rate, and channel condition. The compressed video information is lost due to congestion, channel errors, and transport jitters. The temporal predictive nature of most compression technology causes the undesirable effect of error propagation.

Manuscript received August 22, 2001; revised April 15, 2002. This work was supported in part by the National Science Council of the Republic of China under Grant NSC 90-2213-E-009-139.

H.-C. Huang and T. Chiang are with the Department and Institute of Electronics Engineering, National Chiao Tung University (NCTU), Hsinchu 30050, Taiwan, R.O.C. (e-mail: sleeping.ee89g@nctu.edu.tw; tchiang@cc.nctu.edu.tw).

C.-N. Wang is with the Department and Institute of Computer Science and Information Engineering, National Chiao Tung University (NCTU), Hsinchu 30050, Taiwan, R.O.C.

Publisher Item Identifier 10.1109/TCSVT.2002.800314.

To address the broadcast or Internet multicast applications, the MPEG-4 committee further develops the fine granularity scalability (FGS) profile [1] that provides a scalable approach for streaming video applications. The MPEG-4 FGS representation starts by separating the video frames into two layers with identical spatial resolutions, which are referred to as the base layer and the enhancement layer. The bitstream at the base layer is coded by a nonscalable MPEG-4 advanced simple profile (ASP) while the enhancement layer is obtained by coding the difference between the original DCT coefficients and the coarsely quantized base-layer coefficients in a bitplane-by-bitplane fashion [2]. The FGS enhancement layer can be truncated at any location, which provides fine granularity of reconstructed video quality proportional to the number of bits actually decoded. There is no temporal prediction for the FGS enhancement layer, which provides an inherent robustness for the decoder to recover from any error. However, the lack of temporal dependency at the FGS enhancement layer decreases the coding efficiency as compared to that of the single-layer nonscalable scheme defined in [3].

To improve the MPEG-4 FGS, a motion compensation (MC) based FGS technique (MC-FGS) with a high-quality reference frame was proposed to remove the temporal redundancy for both the base and enhancement layers [5]. The advantage of MC-FGS is that it can achieve high compression efficiency close to that of the nonscalable approach in an error-free transport environment. However, the MC-FGS suffers from the disadvantage of error propagation or drift when part of the enhancement layer is corrupted or lost. Similarly, the PFGS [4] improves the coding efficiency of FGS and provides means to alleviate the error drift problems simultaneously. To remove the temporal redundancy, the PFGS adopts a separate prediction loop that contains a high quality reference frame where partial temporal dependency is used to encode the enhancement-layer video. Thus, the PFGS trades coding efficiency for certain level of error robustness. In order to address the drift problem, the PFGS keeps a prediction path from the base layer to the highest bitplanes at the enhancement layer across several frames to make sure that the coding schemes can gracefully recover from errors over a few frames. The PFGS suffers from loss of coding efficiency whenever a lower quality reference frame is used. Such disadvantageous situation occurs when only a limited number of bitplanes are used or a reset of the reference frame is invoked.

To prevent the error propagation due to packet loss in a variable bit rate channels, the leaky prediction technique was used for the interframe loop in DPCM and subband coding systems [6]–[8]. Based on a fraction of the reference frame, the prediction is attenuated by a leak factor of value between zero and

unity. The leaky prediction strengthens the error resilience at the cost of coding efficiency since only part of the known information is used to remove the temporal redundancy. For a given picture activity and bit error rate (BER), there exists an optimal leak factor to achieve balance between coding efficiency and error robustness [7]. In this paper, we propose a flexible FGS framework that allows encoder to select a tradeoff that simultaneously improves the coding efficiency and maintains adequate video quality for varying bandwidth or error-prone environments.

The rest of this paper will be organized as follows. Section II introduces the basic idea of the robust FGS (RFGS) framework. In Section III, we show the encoder and decoder structures based on the RFGS scheme, and the rate control scheme in the streaming server is explained. The approaches for selecting the optimized parameters are described in Section IV. Section V shows the performance and robustness of the RFGS algorithm based on several typical channel transmission scenarios. Finally, conclusions are given in Section VI.

II. Prediction Techniques for the Enhancement Layer

The MPEG-4 FGS compresses the enhancement layer with only the prediction that comes from the base layer of the current frame. Therefore, truncation of the enhancement layer does not cause error propagation. While providing flexibility in adapting the bandwidth variations and providing robustness to packet loss and errors, the MPEG-4 FGS is worse in coding efficiency as compared to the traditional two-layer SNR scalable scheme because the SNR scalable approach uses a high-quality reference frame. Such an improved coding efficiency comes with a penalty in error propagation whenever there is a loss at the enhancement layer. The picture quality will drift until the next intra-coded frame [4]. Thus, the MPEG-4 FGS approach offers the best error robustness while the SNR scalable approach provides the best coding efficiency. We will describe a novel and flexible framework, which is referred to as RFGS that aims to strike a balance between these two approaches. The RFGS focuses on constructing a better reference frame based on two MC prediction techniques: leaky and partial predictions.

A. Leaky Prediction

The leaky prediction [7] technique scales the reference frame by a factor α, where $0 \leq \alpha \leq 1$ as the prediction for the next frame. The leak factor is used to speed up the decay of error energy in the temporal directions. In RFGS, we use the leak factor to scale a picture that is constructed based on the concept of partial prediction as detailed in the next subsection.

B. Partial Prediction

As described in Fig. 1, the RFGS is constructed with two prediction loops for the base and enhancement layers. The base-layer loop is coded with a nonscalable approach for all frames F_i. The enhancement-layer loop uses an improved quality reference frame that combines the base-layer reconstructed image and partial enhancement layer. Thus, the enhancement-layer loop can be built with an adaptive selection of number of bitplanes for the reference picture. The combinations of selections for each frame constitute multiple prediction paths.

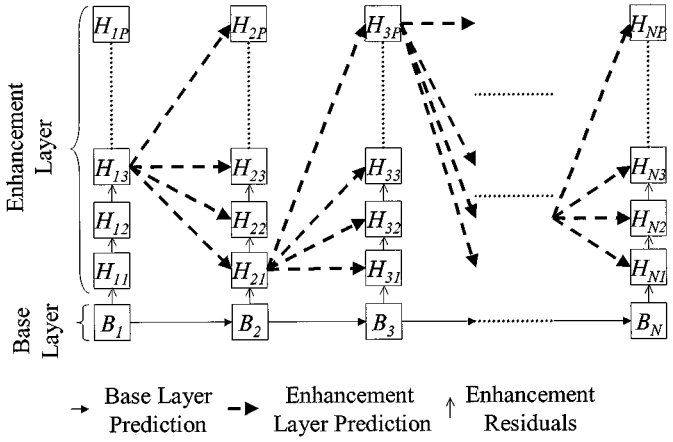

Fig. 1. Partial inter-prediction mode for coding the bitplanes at the enhancement layer using the RFGS coding framework. Each frame has the flexibility to select the number of bitplanes used to generate the high-quality reference frame. For example, the first frame uses three bitplanes to compute the high-quality reference frame.

Let us assume that each frame has P maximal number of bitplanes for the enhancement layer. As the number of bitplanes (denoted as β) used is increased, the residuals will be decreased that translates into improved coding efficiency. On the other hand, the reconstruction errors will accumulate and propagate if the bitplanes used for the reference frame are not available at the decoder. Thus, the parameter β can be used to control the tradeoff between coding efficiency and error robustness.

Combining the concepts of partial and leaky predictions, the first β bitplanes will be scaled by a leak factor. Consequently, if any information at the first β bitplanes is lost, the error will be attenuated by α times for each frame at the enhancement layer. Since the value of α is smaller than unity, the drift will be eliminated in a few frames. Thus, the RFGS is implemented by defining a set of the parameters for each frame t

$$\{M_t(\alpha, \beta)\}, \qquad t = 0, \ldots, (N-1) \tag{1}$$

where the parameter α denotes the leak factor and the parameter β denotes the number of the bitplanes used to construct the reference frame. The symbol N is the total number of frames in the video sequence. As compared to the PFGS [4], the periodic reset of the reference frames can be simulated with a periodic selection of the parameter α as zeros. The MPEG-4 FGS is equivalent to the case of setting α to zero through the whole sequence. As compared to the MC-FGS [5], the use of high-quality reference frames can be simulated with α equals to unity for all reference frames. Thus, the RFGS provides a flexible MC prediction scheme that can be adapted to achieve various tradeoffs as proposed by PFGS [4] and MC-FGS [5].

C. Adaptive Mode Selection

We can easily construct a trellis of predictions based on the selected parameters α and β for each frame. The RFGS leaves great flexibility to optimize the selection of (α, β) to achieve adequate performance in terms of coding efficiency and error robustness. The design is constrained by several parameters such as average bit rate, average BER, and desired video quality. For

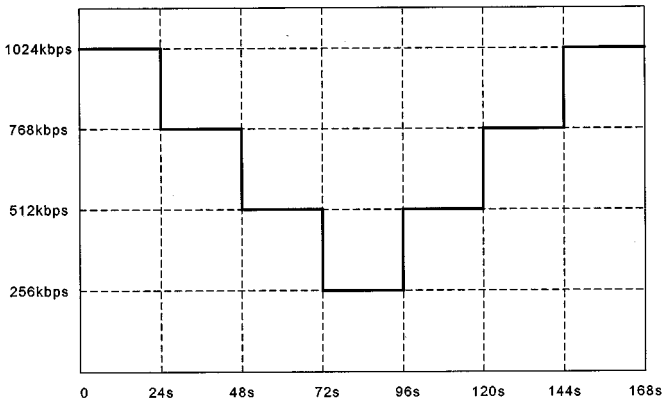

Fig. 2. Channel bandwidth variation pattern for the dynamic test defined in the MPEG document m8002 [12].

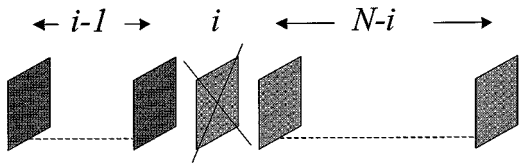

Fig. 3. A transmission scenario with corrupted or lost frame for a video stream of N frames, where the enhancement layer of the ith frame is assumed to be lost.

instance, we have a sample traffic pattern that has significant variation in bandwidth and occasional packet loss, as illustrated in Fig. 2. If a specific traffic pattern is known beforehand, the optimal set of β should match the instantaneously available bandwidth and the drift is nonexistent. However, it is unrealistic to know this traffic pattern so this solution will not be optimal for other traffic patterns. Thus, the RFGS need to select a set of parameters $\{M_t(\alpha, \beta)\}$, $t = 0, \ldots, (N-1)$ that maximizes the average coding efficiency over a range of channel bandwidths.

III. RFGS System Architecture

Based on the concepts of leaky and partial predictions, the RFGS encoder and decoder are constructed as illustrated in Figs. 4 and 5 with all the symbols defined in Table I. As compared to the MPEG-4 FGS [1], the RFGS has added only a few modules including MC, DCT/IDCT, and a reference frame buffer to store the high-quality reference frame that is constructed based on the base and enhancement layers. The concept of leaky and partial predictions can be applied to both the base and enhancement layers. We will explain how to realize the leaky prediction at the enhancement layer in detail in Section III-A–C. The identical steps can be applied for the base layer, except that the predicted frames of both layers are stored in two distinct frame buffers.

A. Functional Description

The base layer is encoded with the advanced simple profile (ASP) using a modification of the B pictures. The B-picture is encoded with a high-quality reference frame at the enhancement layer. There is no drift because B-picture is not used for prediction. The enhancement layer is encoded with the MPEG-4 FGS syntax but with the new prediction schemes. The enhancement layer uses the same motion vectors from the base layer. The MC module uses the base-layer motion vectors and the high quality reference frames to generate the high-quality predictions $ELPI$, as shown in Fig. 4. The difference signal $MCFD_{EL}$ for the enhancement layer is obtained by subtracting $ELPI$ from the original signal F. For the P-pictures, the signal $\hat{D}$ is computed by subtracting $\hat{B}$ from the enhancement-layer difference signal $MCFD_{EL}$. As for the I and B pictures, the signal $\hat{D}$ is computed by subtracting $\hat{B}$ from the base-layer difference signal $MCFD_{BL}$. Finally, the signal $\hat{D}$ is encoded with the MPEG-4 FGS syntax to generate the enhancement-layer bitstream.

B. Leaky and Partial Predictions

Now we will describe the technique to generate the high quality reference image using the leaky and partial predictions. The first β bitplanes of the difference signal $\hat{D}$ are combined with the reconstructed base-layer DCT coefficients $\hat{B}$. The resultant signal is transformed back to the spatial domain using IDCT and is added to the enhancement-layer MC prediction $ELPI$. The difference between the high-quality reference frame and the base-layer reconstructed signal B is computed and attenuated by a leak factor α. The base-layer reconstructed signal B is added back before storing back into the frame buffer.

The encoding of B pictures, as shown in Fig. 4, uses the high-quality reference frame as the extended base layer to form the prediction for both the base and enhancement layers. The base-layer difference signal $MCFD_{BL}$ is first quantized to form the B-picture base layer, and the residual (quantization error) is coded as FGS enhancement layer using MPEG-4 FGS syntax. Since the B picture is not used as reference frame, there is no drift. Thus, we can increase the leak factor to achieve better coding efficiency. However, the inclusion of B pictures at the enhancement layer requires an extra frame buffer to achieve the extra coding gain.

Since the difference between the high-quality reconstructed signal and the low-quality reconstructed signal is attenuated by a leak factor α, the attenuated difference and the low-quality reconstructed signals will be summed together to form the high-quality reference image for the next frame. Therefore, the drift or the difference between the encoder and decoder will be attenuated accordingly. If the leak factor is set to zero, the drift will be removed completely, which is exactly how the MPEG-4 FGS works.

The rationale for performing such a complicated and tricky attenuation process in the spatial domain is because in this way the errors can be recursively attenuated for all the past frames. If the attenuation process is only applied for the first few bitplanes of the current VOP, only the errors occurring in the current VOP are attenuated. The errors that occurred earlier are only attenuated once and can still be propagated to the subsequent frames without further attenuation. In our approach, not only are the errors which occurred in the current VOP attenuated, but also all the errors in the earlier frames are attenuated. After several iterations, the errors will be reduced to zero.

C. Analysis of Error Propagation

The RFGS framework is constructed based on the well-known concept of leaky prediction to improve the error recovery capability as proposed in several other video coding techniques, such as DPCM and the subband video coding in

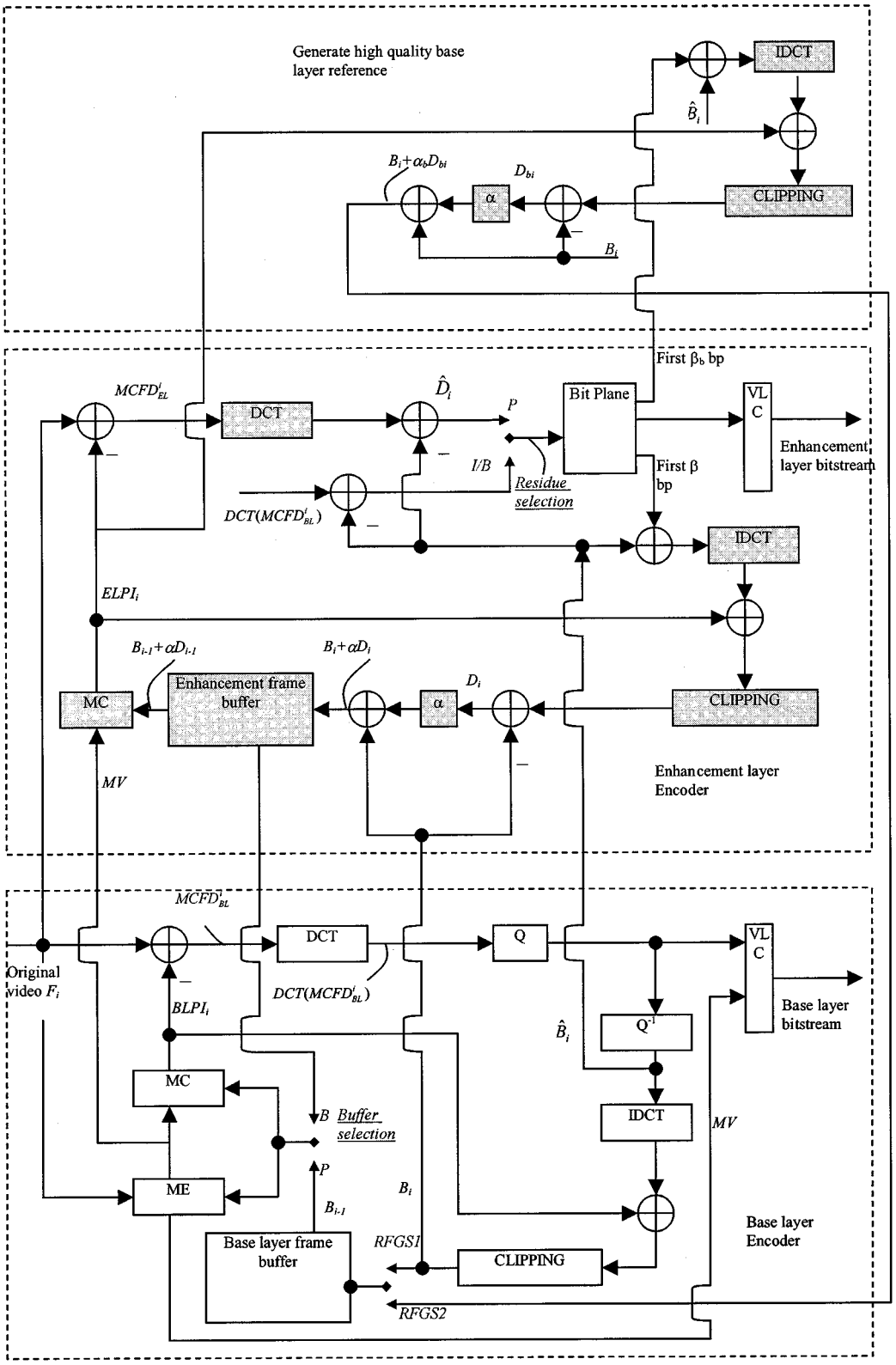

Fig. 4. Diagram of the RFGS encoder framework. The shadowed blocks are the new modules for RFGS as compared to MPEG-4 baseline FGS.

[6]–[8]. The major distinction in our approach is the technique to compute the reference frame and the final residual for transmission. In the RFGS framework, the high-quality reference frame consists of three components, including the MC base-layer reconstructed frame, the quantized difference signal of the base layer, and the attenuated final residual at the enhancement layer. Thus, we have the following relationship:

$$\text{High quality reference image} = B + \alpha \times D$$

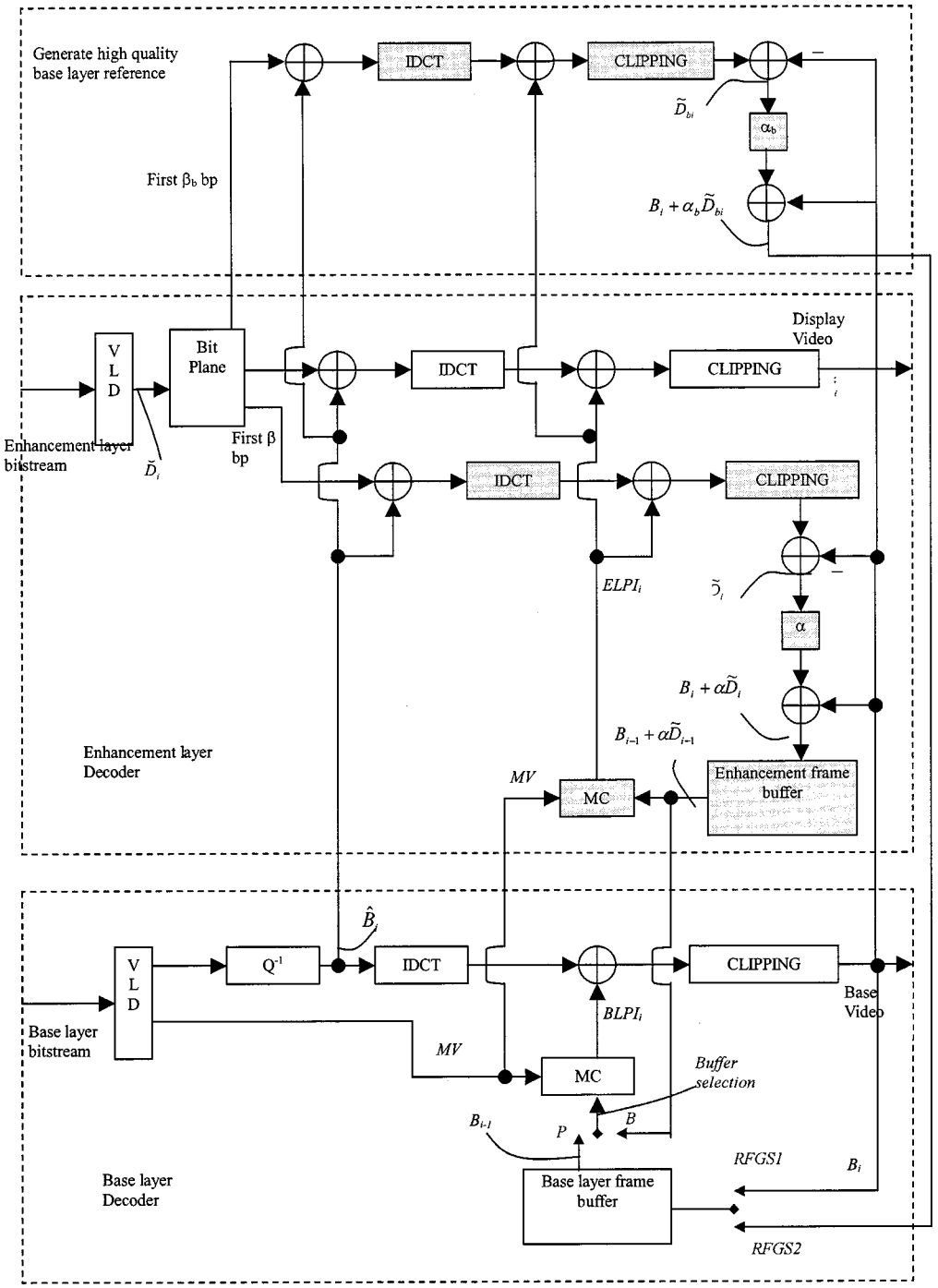

Fig. 5. Diagram of the RFGS decoder framework. The shadowed blocks are the new modules for RFGS as compared to MPEG-4 baseline FGS.

where B is the base-layer reconstructed signal and D is the final residual used at the enhancement layer.

We now compute the reconstruction errors when only partial bitstream is available. As illustrated in Fig. 4, we describe the technique to form the base and enhancement layers. For the current frame, the original frame at time i is denoted as F_i. At the base layer, the reconstructed frame of the previous time $i-1$ is denoted as B_{i-1}. The base layer MC frame difference signal is denoted as $MCFD^i_{BL}$ at time i. Thus, the original frame at time i can be computed as

$$F_i = (B_{i-1})_{\rm mc} + MCFD^i_{BL}. \tag{2}$$

The subscript mc means that the $(B_{i-1})_{\rm mc}$ is the MC version of B_{i-1}. That is, the $(B_{i-1})_{\rm mc}$ equals the $BLPI_i$, as illustrated in Fig. 4

$$BLPI_i = (B_{i-1})_{\rm mc}. \tag{3}$$

The coded version of the base-layer difference signal $MCFD^i_{BL}$ is denoted as frame $\hat{B}_i$. Let the quantization error after encoding be Q_i. The relationship between $MCFD^i_{BL}$, $\hat{B}_i$, and Q_i is

$$MCFD^i_{BL} = \hat{B}_i + Q_i. \tag{4}$$

TABLE I
TERMINOLOGY OF THE RFGS CODING FRAMEWORK

Notation	Definitions
F	The original image
BLPI	Predicted base layer frame that is generated by motion compensation from the base layer frame buffer.
MCFD_{BL}	Motion compensated frame difference of the base layer, which is the difference between BLPI and the original image.
$\hat{B}$	Coded DCT coefficients of frame $MCFD_{BL}$. The $\hat{B}$ before de-quantization will be compressed as the base layer bitstream.
B	The base layer reconstructed image, which is the summation of *BLPI* and $\hat{B}$. B will be stored in the base layer frame buffer.
ELPI	Predicted frame of the enhancement layer that is generated by motion compensation from the enhancement layer frame buffer.
$MCFD_{EL}$	Motion compensated frame difference of the enhancement layer which the difference between *ELPI* and the original image.
$\hat{D}$	Difference signal between $MCFD_{EL}$ and $\hat{B}$ for *P*-pictures or MCFD_{BL} and $\hat{B}$ for *I*-pictures and *B*-pictures. $\hat{D}$ will be compressed as the enhancement layer bitstream.
D	The final residual used at the enhancement layer prediction loop in the encoder. $(B+\alpha D)$ will be stored at the enhancement layer frame buffer of the encoder.
$\breve{D}$	The received $\hat{D}$ in the decoder side. Since there may be truncation or error during the transmission of enhancement layer bitstream, $\hat{D}$ and $\breve{D}$ may be different.
$\Delta\hat{D}$	The difference between $\hat{D}$ and $\breve{D}$.
$\tilde{D}$	The reconstructed D in the decoder side. $(B+\alpha\tilde{D})$ will be stored at the enhancement layer frame buffer of the decoder.

The quantized version of the difference signal $MCFD^i_{BL}$, which equals to the signal $\hat{B}_i$ before de-quantization, is compressed as the base-layer bitstream. In the MPEG-4 FGS coding scheme, the quantization error Q_i will be encoded to generate the enhancement-layer bitstream.

For the enhancement layer, the base-layer reconstructed frame B_{i-1} of the previous time $i-1$ and αD_{i-1} will be summed to create the high-quality reference frame, where D_{i-1} is the actual information used from the enhancement layer of the previous frame at time $i-1$. After MC, the $MCFD^i_{EL}$ is computed from

$$F_i = (B_{i-1} + \alpha D_{i-1})_{\text{mc}} + MCFD^i_{EL} \tag{5}$$

where the $(B_{i-1} + \alpha D_{i-1})_{\text{mc}}$ is the same as the $ELPI_i$ in Fig. 4. That is

$$ELPI_i = (B_{i-1} + \alpha D_{i-1})_{\text{mc}}. \tag{6}$$

Assume that there is redundancy between $MCFD^i_{EL}$ and $\hat{B}_i$ (the coded version of $MCFD^i_{BL}$), the frame $\hat{B}_i$ is subtracted from the difference signal $MCFD^i_{EL}$ to remove such redundancy. The resultant difference is denoted as $\hat{D}_i$, which will be compressed for transmission at the enhancement layer. Thus, we have

$$\hat{D}_i = MCFD^i_{EL} - \hat{B}_i. \tag{7}$$

Substitute (7) into (5), and the original image F_i can be reformulated as

$$F_i = (B_{i-1} + \alpha D_{i-1})_{\text{mc}} + \hat{B}_i + \hat{D}_i. \tag{8}$$

By grouping the base and enhancement-layer information, (7) becomes

$$\begin{aligned} F_i &= (B_{i-1})_{\text{mc}} + \hat{B}_i + (\alpha D_{i-1})_{\text{mc}} + \hat{D}_i && (9)\\ &= B_i + D_i && (10)\end{aligned}$$

where

$$B_i = (B_{i-1})_{\text{mc}} + \hat{B}_i \tag{11}$$

and

$$D_i = (\alpha D_{i-1})_{\text{mc}} + \hat{D}_i. \tag{12}$$

The signals B_i and D_i will be used for the prediction of next frame. It should be noted that for simplicity, we assume all of the bitplanes in $\hat{D}_i$ are used at the enhancement-layer prediction loop.

By expanding the recursive formula of D_i in (12), we can get

$$\begin{aligned} D_i &= (\alpha((\alpha D_{i-2})_{\text{mc}} + \hat{D}_{i-1}))_{\text{mc}} + \hat{D}_i \\ &= (\alpha((\alpha((\alpha D_{i-3})_{\text{mc}} + \hat{D}_{i-2}))_{\text{mc}} + \hat{D}_{i-1}))_{\text{mc}} + \hat{D}_i \\ &= \cdots. \end{aligned} \tag{13}$$

As demonstrated in (13), it is obvious that any errors in the final residual D_i will be attenuated in the RFGS framework. Assume there is a network truncation or error at the enhancement layer for frame F_{i-2}, we denote the received enhancement-layer bitstream as $\breve{D}_{i-2}$ and the transmission error is denoted as $\Delta\hat{D}_{i-2}$. Thus, we have

$$\hat{D}_{i-2} = \breve{D}_{i-2} + \Delta\hat{D}_{i-2} \tag{14}$$

and the reconstructed version of D_{i-2} is denoted as $\tilde{D}_{i-2}$. Thus

$$\begin{aligned} \tilde{D}_{i-2} &= (\alpha D_{i-3})_{\text{mc}} + \breve{D}_{i-2} \\ &= (\alpha D_{i-3})_{\text{mc}} + \hat{D}_{i-2} - \Delta\hat{D}_{i-2}. \end{aligned} \tag{15}$$

Comparing (12) and (15), the difference between D_{i-2} and $\tilde{D}_{i-2}$ is $\Delta\hat{D}_{i-2}$.

Now we trace back to the frame F_{i-1}. For simplicity, we assume that there is no error or bit truncation at the enhancement layer for frames F_{i-1} and F_i. Expanding (15), we have

$$\begin{aligned} \tilde{D}_{i-1} &= \left(\alpha\tilde{D}_{i-2}\right)_{\text{mc}} + \hat{D}_{i-1} \\ &= \left(\alpha\left((\alpha D_{i-3})_{\text{mc}} + \hat{D}_{i-2} - \Delta\hat{D}_{i-2}\right)\right)_{\text{mc}} + \hat{D}_{i-1}. \end{aligned} \tag{16}$$

The difference between D_{i-1} and $\tilde{D}_{i-1}$ is now $\alpha(\Delta\hat{D}_{i-2})$.

Now we move on to the frame F_i and get

$$\begin{aligned} \tilde{D}_i &= \left(\alpha\tilde{D}_{i-1}\right)_{\text{mc}} + \hat{D}_i \\ &= \left(\alpha\left(\left(\alpha\left((\alpha D_{i-3})_{\text{mc}} + \hat{D}_{i-2} - \Delta\hat{D}_{i-2}\right)\right)_{\text{mc}} \right.\right. \\ &\qquad \left.\left. + \hat{D}_{i-1}\right)\right)_{\text{mc}} + \hat{D}_i. \end{aligned} \tag{17}$$

The difference between D_i and $\tilde{D}_i$ is now $\alpha^2(\Delta\hat{D}_{i-2})$.

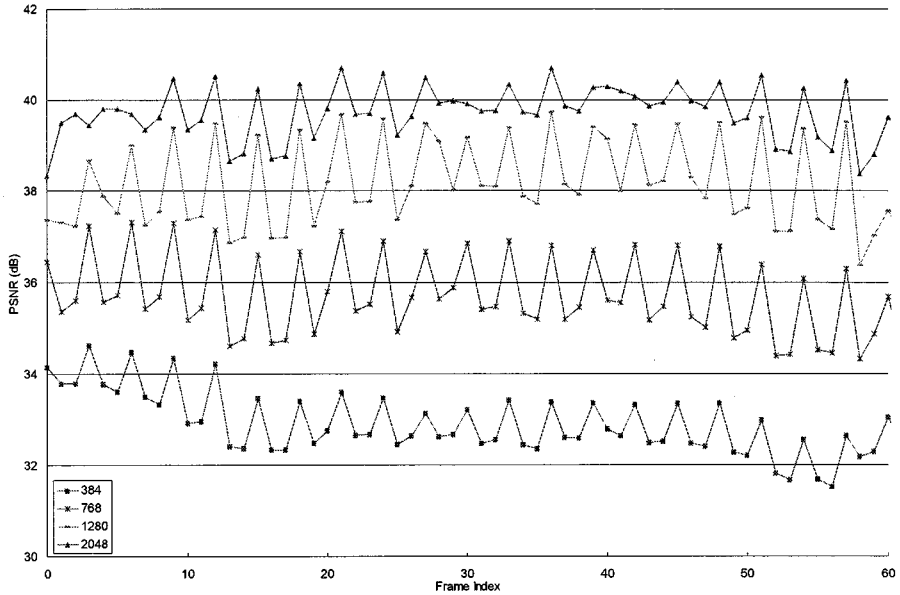

Fig. 6. Visual qualities of the reconstructed pictures using the proposed RFGS rate control scheme. We provide the quality of the first 60 frames of the Foreman bitstream. The base-layer bitstream is encoded with a bit rate of 256 kbps. The enhancement-layer bitstream is truncated at several bit rates to understand the variation in PSNR for various channel bandwidths. The results show that the PSNR variation is smaller than 2 dB at various bit rates.

From the above derivations, it is obvious that the errors occurred in the decoded bitstream at the enhancement layer will be attenuated by a factor of α for each iteration. After several iterations, the error will be attenuated to zero for α less than unity. Thus, the drift is removed from the system.

As an example shown in Fig. 3, there is a video bitstream for N frames. Let us assume that only the ith frame F_i is lost during transmission, the mean square error for the reconstructed enhancement-layer frame of size $H \times M$ can be computed as

$$e_i^2 = \frac{1}{HM} \sum_{x=1}^{H} \sum_{y=1}^{M} \left(\hat{F}_i(x, y) - \hat{F}_i^e(x, y) \right)^2 \qquad (18)$$

where the signal $\hat{F}_i(x, y)$ represents the reconstructed frame with all bitplanes and the $\hat{F}_i^e(x, y)$ represents the reconstructed frame where some bitplanes are lost. Consequently, the average video quality degradation of the reconstructed picture that is caused by the errors at frame F_i is

$$\begin{aligned} \Delta MSE_{\text{avg}} &= \frac{\left(1 + \alpha^2 + \cdots + \alpha^{2(N-i)}\right)}{N} e_i^2 \\ &= \frac{1 - (\alpha^2)^{N-i+1}}{(1 - \alpha^2)N} e_i^2. \end{aligned} \qquad (19)$$

As α tends to unity, the average MSE accumulated through the prediction loop will accumulate as expected. For the leak factor less than unity, the degradation will be decreased exponentially as shown in Fig. 15. The error attenuation can be approximated with an exponential function

$$\Delta PSNR(\alpha) = K_1(\alpha) e^{-K_2(\alpha) t} = K_1(\alpha) e^{-(t/\tau(\alpha))} \qquad (20)$$

where $K_1(\alpha)$ and $K_2(\alpha)$ are constants that vary as a function of α and can be computed using the least-square approximation technique. The constant $K_2(\alpha)$ is a reciprocal of the time constant $\tau(\alpha)$ for an exponential function. It is expected that $K_2(\alpha)$ is increased as α is decreased because the errors are attenuated faster when α is decreased. As demonstrated in Fig. 17, the time constant $\tau(\alpha)$ is reduced by half when the leak factor α is reduced to 0.9. Thus, the selection of the leak factor α is a critical issue to achieve a better balance between coding efficiency and error robustness. For α that is close to unity, the coding efficiency is the best while the error robustness is the worst with longest attenuation time constant. On the other hand, for α that is close to zero, the error recovery property will be enhanced at the cost of less coding efficiency.

D. High-Quality Reference in Base Layer

As mentioned in Section III-A, the signal $\hat{D}$, which is transmitted at the enhancement layer, is computed by subtracting $\hat{B}$ from the enhancement-layer difference signal $MCFD_{EL}$. Such a difference reduces the energy of the residuals but increases the dynamic range of the signal $\hat{D}$, which is particularly inefficient for bitplane coding [9]. Thus, there is room for further improvement. Additionally, there is redundancy that exists between the high-quality reference image for the enhancement-layer and the base-layer difference signal $MCFD_{BL}$. To decrease the fluctuation of $\hat{D}$ and remove the said redundancy, a higher quality reference image for the base layer is used. As compared to the signal B, the statistical characteristics of the higher quality reference for the base layer is closer to that of the high quality reference image for the enhancement layer. Therefore, the dynamic range of $\hat{D}$ is reduced and the temporal redundancy between the high-quality reference image for the enhancement layer and the signal $MCFD_{BL}$ is also reduced.

In Figs. 4 and 5, we illustrate how the high-quality reference is generated for the base layer. Part of the enhancement layer is duplicated in the part "generate high-quality base-layer reference" to form the high-quality reference image for the base layer. The derivation of the high-quality reference image for the base layer is identical to that for the enhancement layer, except that the base layer has its own RFGS parameters, which are denoted as α_b and β_b, respectively. The resultant high-quality reference image will replace the signal B and is stored in the base-layer frame buffer.

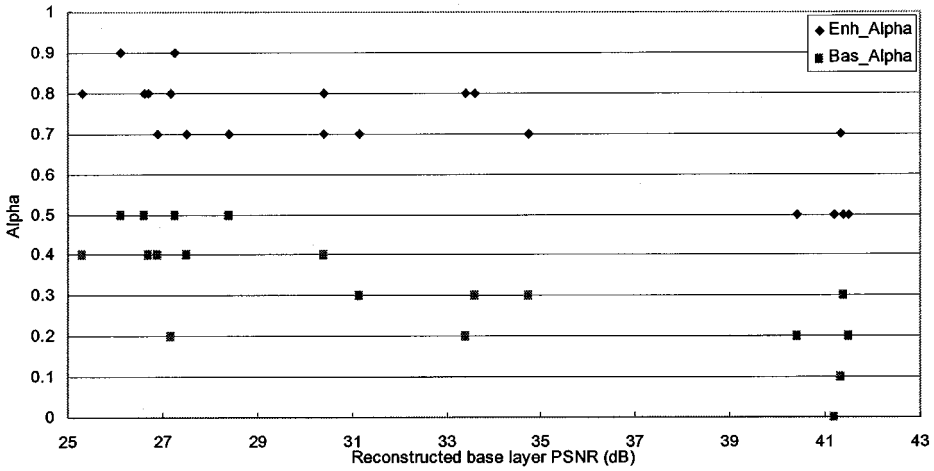

Fig. 7. Linear dependency between near-optimal leak factor and the picture quality in PSNR of the base layer. The frames within five GOV's, where each has 60 frames, are used for the simulations with the four sequences, namely Akiyo, Carphone, Foreman, and Coastguard.

Although the use of a high-quality reference image for the base layer can achieve a better coding efficiency, it suffers from drift problem at low bit rate [5]. The drift at the base layer cannot be removed because the base-layer reference image is not attenuated by α. To strike a balance between the coding efficiency and the error drift, a small α should be used for the base layer. With a suitable selection of α_b, the drift at low bit rate can be reduced and the coding efficiency is significantly enhanced for medium and high bit rates.

E. Rate Control for the Enhancement Layer

For the MPEG-4 FGS, the rate control is not an issue since there is no temporal dependency among frames at the enhancement layer. However, the rate control is relevant in the case of the RFGS, especially when the expected range of bandwidth in operation is widely varied. The server can adaptively determine the number of bits to be sent frame by frame. When the expected channel bandwidth is small, the bitplanes that are used to construct the high quality reference frame may not be available mostly. Since only the I-picture and P-pictures are used as the reference frames, the limited bandwidth should be allocated to those anchor frames at low bit rate [5]. The B-pictures will also be improved because better anchor frames are used for interpolation. When the average bit rate becomes higher, additional bits should be allocated to B pictures, where bits can be spent on the most significant bitplanes for more improvements. By allocating more bits to the P pictures the overall coding efficiency is improved but the PSNR values vary significantly between the adjacent P picture and B picture, especially at a medium bit rate, where most of the bitplanes in P pictures have been transmitted but only a few bitplanes for B pictures are transmitted. The maximal PSNR difference may be up to 4 dB in our simulation. To achieve better visual quality, as shown in Fig. 6, the proposed rate control scheme reduces the variance of the PSNR values of the adjacent pictures at the cost of decreasing the overall quality by about 0.5 dB in PSNR. Since the RFGS scheme provides an embedded and fully scalable bitstream, the proposed rate control can occur at server, router, and decoder. In this paper, we perform the rate control at the server side for all simulations.

IV. Selection of the RFGS Parameters

A. Selection of the Leaky Factor

In order to find an algorithm that computes the optimized α, we perform a near-optimal exhaustive search for the parameters by dividing every sequence into several segments that contain a group of video object planes (GOV). In our simulation, each GOV has 60 frames. The "near optimal" scenario is defined based on the proposed criterion of the "average weighted difference" (AWD), which is the weighted sum of the PSNR differences between the RFGS and the single-layer approaches for a given bit rate range. Thus

$$AWD = \sum_{BR} W(BR) \times D(BR) \tag{21}$$

where BR is a set of evenly spaced bit rates for a given bit rate range. The symbol $W(BR)$ is the weighting function for the bit rate set BR. $D(BR)$ is a set of the PSNR differences between the RFGS and single-layer approaches for every bit rate from the set BR. In our simulations, the set BR is defined by

$$BR = \{256, 512, 768, 1024, 1280, 1536, 1792, 2048, 2304\} \text{ kbps}$$

and the weighting function is

$$W(\cdot) = \{2, 2, 2, 2, 1, 1, 1, 1, 1\}$$

where the importance of the PSNR differences at low bit rate is stressed.

To observe the influence of the leak factors on the coding efficiency, the bitplane numbers for both layers are fixed at three bitplanes. The parameters α_e and α_b are scanned from 0.0 to 0.9 with a step size of 0.1. All the combinations of α_e and α_b are employed for each GOV within the sequence and the pair of α_e and α_b with minimal AWD is selected. Thus, we can get a near-optimal combination of α_e and α_b for each GOV. The results would be optimal if we adapt α_e and α_b at frame level but the complexity is prohibitive.

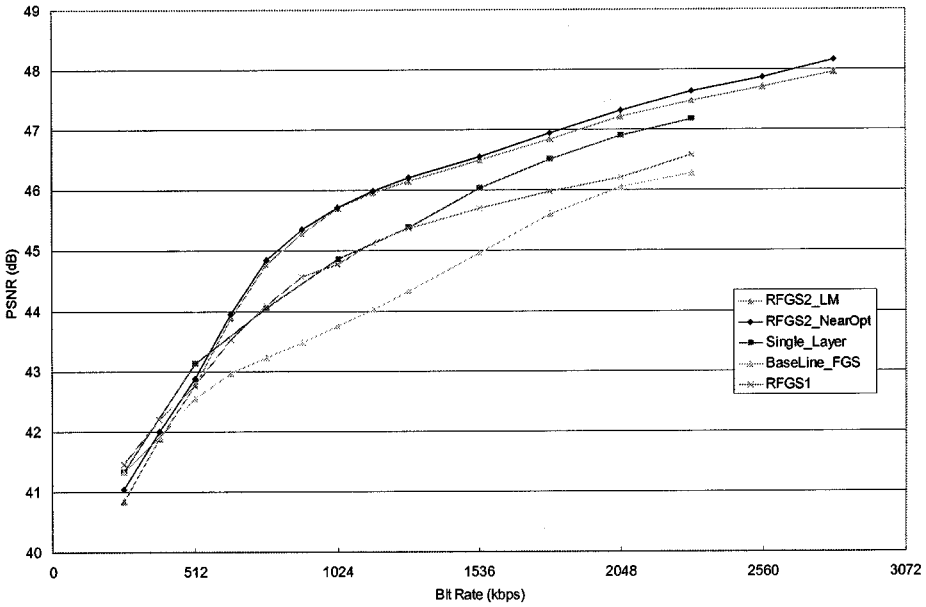

Fig. 8. PSNR versus bit rate comparison between FGS, RFGS, and single-layer coding schemes for the Y component of the Akiyo sequence, where β is 3. We use three different coding schemes including "RFGS1," "RFGS2_NearOpt," and "RFGS2_LM" in the experiments. "RFGS1" uses the RFGS algorithm for the enhancement-layer only. "RFGS2" uses the RFGS algorithm for both the enhancement and the base layers. "NearOpt" means the result of the near-optimal approach and "LM" means the results using the proposed linear model.

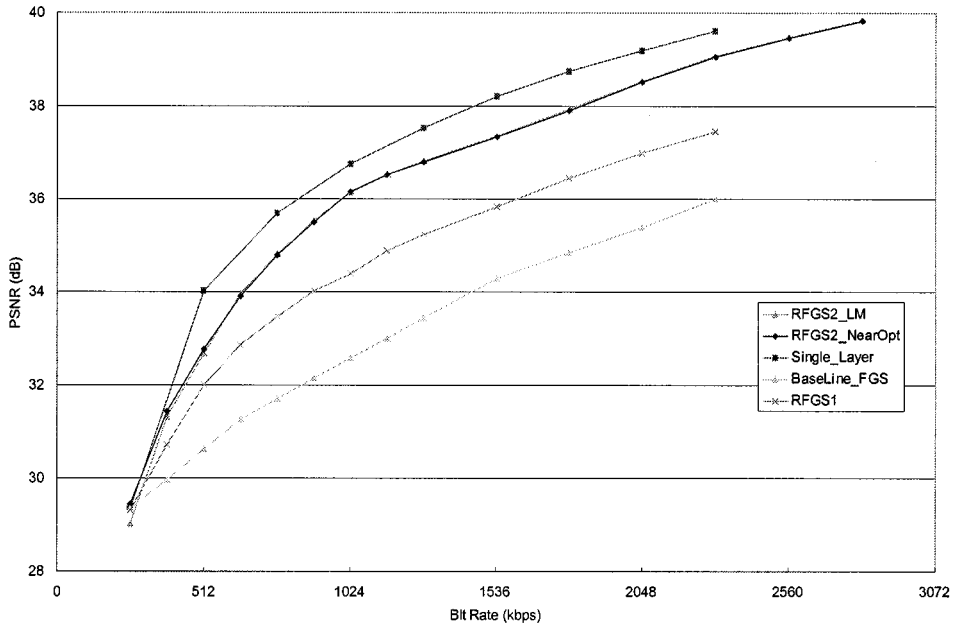

Fig. 9. PSNR versus bit rate comparison between FGS, RFGS, and single-layer coding schemes for the Y component of the Foreman sequence, where β is 3. We use three different coding schemes including "RFGS1," "RFGS2_NearOpt," and "RFGS2_LM" in the experiments. "RFGS1" use the RFGS algorithm for the enhancement layer only. "RFGS2" uses the RFGS algorithm for both the enhancement and base layers. "NearOpt" means the result of the near-optimal approach and "LM" means the results using the proposed linear model.

In Fig. 7, we show the relationship between the near-optimal combinations of α_e and α_b and the base-layer PSNR values with the experimental results using four sequences based on the GOV-based scheme. As the PSNR value of the base-layer reconstructed frame is decreased, the near optimal α tends to be increased accordingly. Their relationship is almost linear if we eliminate several outliers, which provides a linear model for computing the near-optimal α based on the PSNR value of the base layer. For each frame, we first get the base-layer PSNR values after encoding. Based on the derived PSNR value per frame and the proposed linear model, we compute both α_e and α_b and encode every frame at the enhancement layer. From Figs. 8–10, we find that the RFGS using the linear model has almost identical PSNR values as the RFGS based on the near optimal exhaustive search, which has at maximum a 0.2-dB difference. The performance of the RFGS based on the proposed linear model is much superior to the RFGS with fixed α_e and α_b found empirically.

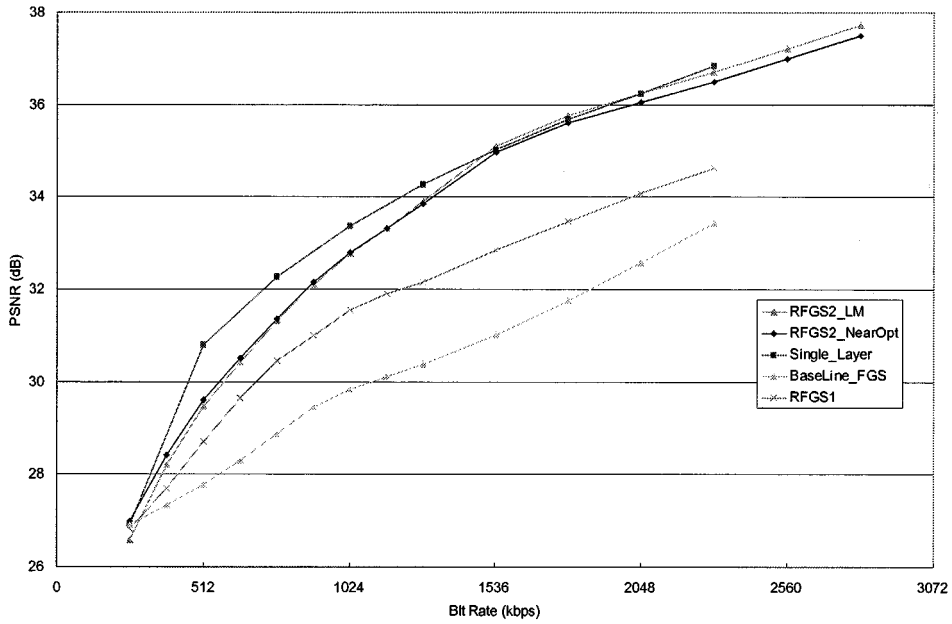

Fig. 10. PSNR versus bit rate comparison between FGS, RFGS, and single-layer coding schemes for the Y component of the Coastguard sequence, where β is 3. We use three different coding schemes including "RFGS1," "RFGS2_NearOpt," and "RFGS2_LM" in the experiments. "RFGS1" use the RFGS algorithm for the enhancement layer only. "RFGS2" uses the RFGS algorithm for both the enhancement and base layers. "NearOpt" means the result of the near-optimal approach and "LM" means the results using the proposed linear model.

B. The Number of Bitplanes

Similarly, we can encode video sequences using different combinations of enhancement layer β and base layer β (denoted as β_e and β_b, respectively), where α_e and α_b are computed with the proposed linear model. Empirically, we find that performance is better when 2–4 bitplanes are used for coding. By applying all possible combination of β_e and β_b within a specified range to the whole sequence, we found that the coding efficiency with identical β for both layers is better than that with distinct β for each layer. The optimal β can be selected based on the range of the target bandwidth. When the target bandwidth is smaller than 512 kbps, the experiments in Fig. 11 show that the RFGS with $\beta = 2$ has the best performance. When the bandwidth is from 256 kbps to 1.2 Mbps, the RFGS with $\beta = 3$ provides the maximal gain in PSNR for most bit rates. When the bandwidth is even higher, the RFGS takes 4 bitplanes to achieve the optimal average coding efficiency. Thus, the number of bitplanes is selected based on the target range of the channel bandwidths. Our framework provides a flexible support for all of them.

V. Experiment Result and Analyses

Extensive experiments have been performed to demonstrate the performance of the proposed RFGS coding technique. From Figs. 8–10, the coding efficiency of the RFGS is compared with those of the baseline FGS coding ("Baseline_FGS") and the single-layer nonscalable coding schemes ("Single_layer"). These two techniques are considered as the lower and upper bounds for the performance. There are three different coding schemes for the RFGS. The scheme, labeled as "RFGS1," uses the RFGS algorithm for the enhancement layer only. The other schemes, denoted as "RFGS2_NearOpt" and "RFGS2_LM," adopt the RFGS algorithm for both the enhancement and the base layers simultaneously as mentioned in Section III-D. The "RFGS2_NearOpt" provides the near-optimal results and the "RFGS2_LM" denotes the results by selecting the parameters based on the proposed linear model in the Section IV-A. In Figs. 12 and 13, we compare the performance of the RFGS that selects the leak factor based on the proposed linear model with that of the macroblock-based PFGS [11]. All performance comparisons among the FGS, PFGS, RFGS, and single-layer coding schemes are based on the reconstructed video quality in PSNR for the given bit rate.

A. The Testing Conditions

From Figs. 8–10, we adopt the testing condition B of the core experiments as specified by the MPEG-4 committee [10] and the MPEG-4 reference encoder with the Advanced Simple Profile for the base layer. In these experiments, the three sequences including Akiyo, Foreman, and Coastguard of CIF format are used for testing. For each sequence, every GOV has size of 60 frames that consist of one I-picture, 19 P-pictures, and two B-pictures between each pair of P-pictures. To derive the motion vectors for P-pictures and B-pictures, a simple half-pixel motion estimation scheme using linear interpolation is used. The search range of the motion vectors is set to ± 31.5 pixels. The bit rate of the base layer is 256 kbps with TM5 rate control, and the frame rate is 30 Hz. To simulate the possible channel bandwidth variation, the total bit rate of the enhancement-layer bitstream is truncated to bit rate ranging from 0 to 2048 kbps with an interval of 128 kbps. In each category, a simple frame-level bit allocation with a truncation module is used in the streaming server to obtain optimized quality for the given bandwidth.

For Figs. 12 and 13, we follow the testing condition A and B as described in [11]. The Foreman and Coastguard sequences of

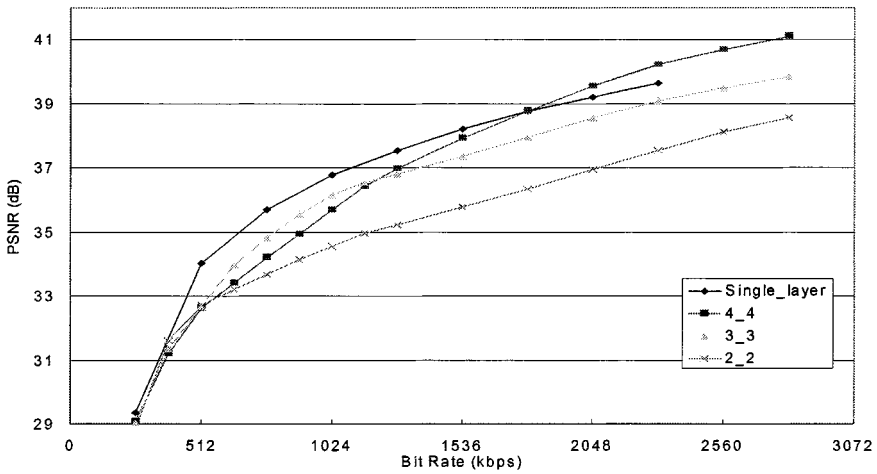

Fig. 11. PSNR versus bit rate comparison between various values of RFSG parameter β for the Y component of the Foreman sequence, where the leak factor α is selected with the proposed linear model.

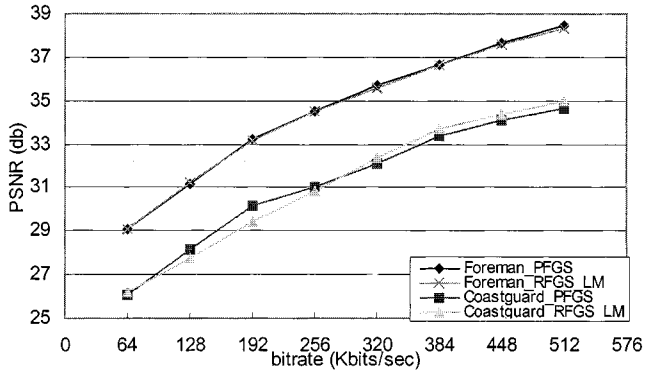

Fig. 12. PSNR versus bit rate comparison between RFGS and PFGS for the Y component of the Coastguard and Foreman sequences in CIF format using the test condition A in the MPEG document m6779 [11]. For RFGS, β is 3.

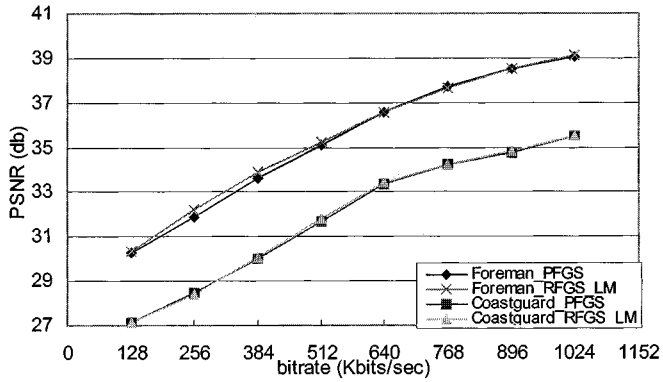

Fig. 13. PSNR versus bit rate comparison between RFGS and PFGS for the Y component of the Coastguard and Foreman sequences in CIF format using the test condition B from the MPEG document m6779 [11]. For the RFGS, β is 3.

CIF format are used for simulation, where only one GOV and no B-picture are used. For the testing condition A, the bit rate of the base layer is 64 kbps and the TM5 rate control is adopted with frame rate of 5 Hz. The enhancement-layer bitstream is truncated to the bit rates ranging from 0 kbps to 448 kbps with an interval of 64 kbps. For the testing condition B, the bit rate of the base layer is 128 kbps and TM5 rate control with frame rate of 10 Hz. The enhancement-layer bitstream is truncated to bit rates ranging from 0 to 896 kbps with an interval of 128 kbps.

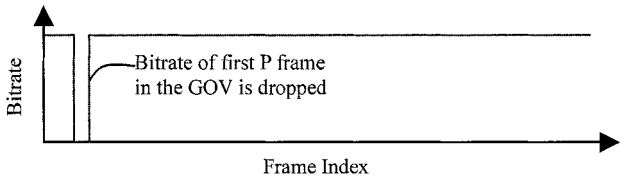

Fig. 14. Sample bandwidth profile to test the error-recovery capability of the RFGS technique.

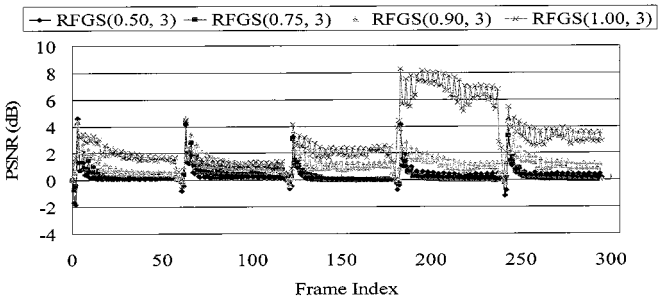

Fig. 15. Error attenuation in PSNR for the Y component of the Akiyo sequence under different α in the RFGS1 framework, where the pair of the values indicates the prediction mode parameters (α, β).

B. Performance Comparisons

For the three specified test sequences, we first show the performance of the RFGS schemes with the GOV structure with B-pictures. From Figs. 8–10, as compared to the baseline FGS, our results show that the RFGS has improved by about 2 dB in PSNR for the fast motion sequences such as Foreman and Coastguard and improves up to 1.1 dB for the slow motion sequence such as Akiyo over the baseline FGS. When the RFGS method labeled as "RFGS2_LM" is applied for both layers, there are up to 3.6 dB and 4.1 dB gain in PSNR over the baseline FGS for the Foreman and Coastguard sequences, respectively. For the Akiyo sequence, the RFGS also has 2.0 dB gain in PSNR over the baseline FGS. To compare with the single-layer approach, the RFGS scheme has 0.6–1.3-dB loss under the various bit rates for the Foreman sequence. For the Coastguard sequence, as compared to the single-layer approach, the RFGS has 1.4-dB loss in PSNR at low bit rate and the almost identical PSNR values at medium and high bit rates. Additionally, the RFGS for the

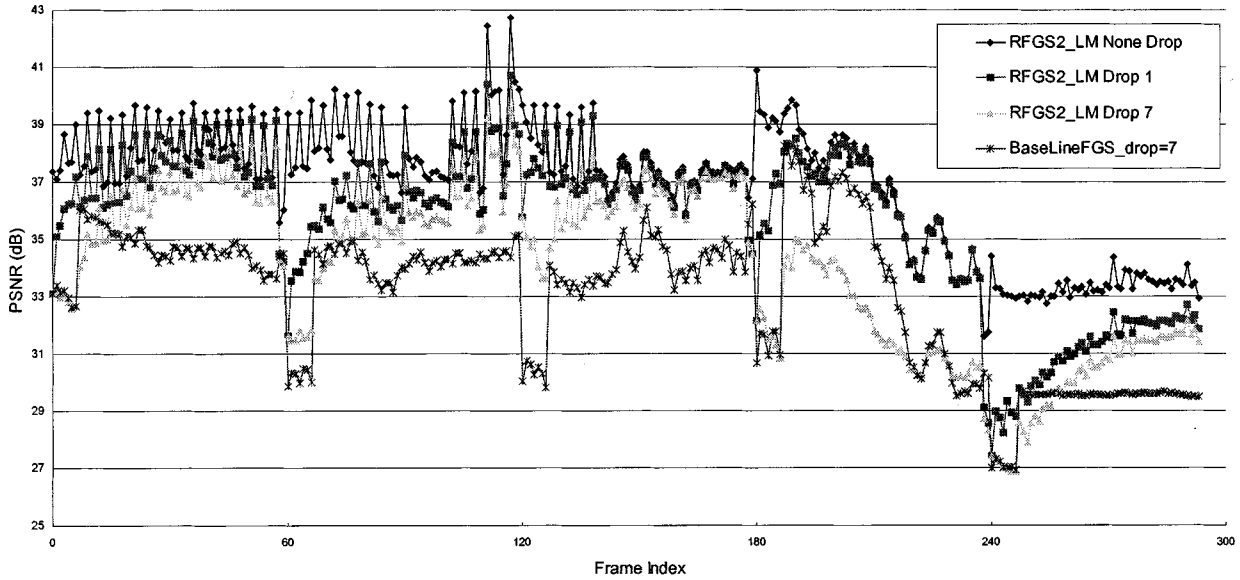

Fig. 16. Error attenuation in PSNR for the Y component of the Foreman sequence using the RFGS2_LM framework. All the curves denote truncation of the enhancement-layer bitstream at 1024 kbps. For the curve labeled "RFGS2_LM Drop 1," the first frame of each GOV is dropped. For the curve labeled "RFGS2_LM Drop 7," the first seven frames of each GOV are dropped. For the curve labeled "RFGS2_LM None Drop," no frame is dropped. The curve labeled "BaseLineFGS_drop $=$ 7" is the baseline FGS with the first seven frames of each GOV dropped.

Akiyo sequence is actually better than the single-layer approach by around 0.3–0.9 dB at medium and high bit rates.

It is interesting that the RFGS2 outperforms the single layer at a high bit rate for the slow motion sequences. For the single-layer approach, only one VLC table is used and it cannot be optimal for a wide range of bit rates. In the FGS approach, however, the different bitplanes have their own VLC tables that can approach to the entropy of the DCT coefficients at both low bit rate and high bit rate. The RFGS2 algorithm removes most of the temporal redundancy and reduces the dynamic range of the residuals. It can encode more efficiently using better VLC tables designed for the high bit rate.

When only the base-layer bitstream is decoded for the extremely low bit-rate case, all three sequences have the PSNR values worse than the PSNR by the single layer by about 0.3–0.5 dB because the RFGS2 uses the enhancement-layer information for the base-layer prediction. Since the there is no leaky factor applied for the base layer, we have error drift even when α_b is small. Considering the significant improvement at the medium and high bit rates, the modest loss of PSNR value at the base layer is acceptable.

Now we compare the results of the RFGS with the macroblock-based PFGS [11] based on the GOV structure without the use of B-pictures and the rate control scheme defined in Section III-E. The experiments show that the error drift for RFGS2 is more serious since all the frames are P-pictures and all of their errors are propagated. Therefore α_b should be set as zero to eliminate the drift at low bit rate. For Figs. 12 and 13, the frame-based RFGS results are quite close to the macroblock-based PFGS [11]. It should be mentioned that identical linear model of the enhancement layer is used to compute α_e.

C. Test for Error Recovery Capability

To verify the error-recovery capability of the RFGS, a simple experiment is performed to demonstrate the worst-case scenario

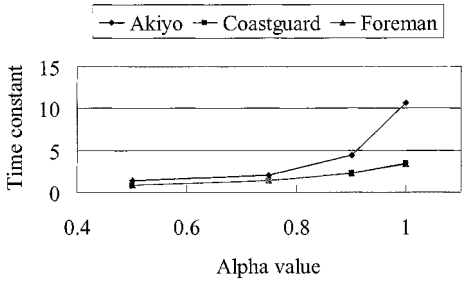

Fig. 17. Relationship between the leak factor α and the time constant τ for the error attenuation. For each curve, β is 3.

when there is bandwidth variation that can cause maximal effect of drift. We assume the network bandwidth is sharply dropped for every first P-picture transmitted of each GOV and the bit budget for the other frames is set as 1024 kbps. Such a bandwidth scenario is illustrated in Fig. 14. Since only the first P-picture for the enhancement layer is lost and the degradation of the subsequent frames will be caused only by the errors from this P picture. The same testing conditions and the video sequences are used as in [10]. To verify the error attenuation of RFGS mentioned in Section II-C, we first examine the RFGS1 method about the speed of the error recovery for various α. In all the simulations, β is set as 3 and α equals to one of the four predefined values, 0.5, 0.75, 0.9, and 1.0. As shown in Fig. 15, the error attenuation capability of the RFGS framework is strongly affected by the value of α used. In the worst-case scenario where no enhancement bit is received, the PSNR loss is more than 5 dB as compared to the PSNR under an error-free condition. For a small α of 0.5, the error is attenuated very fast. For example, in Fig. 15, after fourth P-pictures within the first GOV, the PSNR differences are reduced to about 0.1 to 0.3 dB. When α equals to unity, as shown in the fourth GOV in Fig. 15, the drift lasts for a long time. We provide the performance of RFGS2_LM under the burst error in Fig. 16. We simulate the

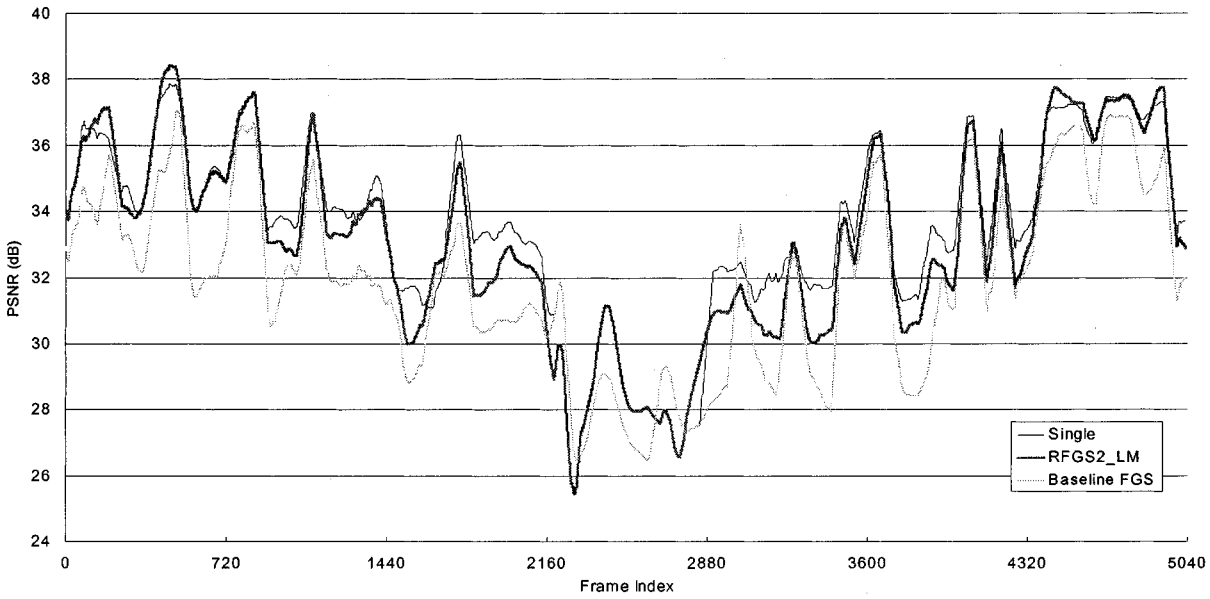

Fig. 18. Comparison of visual quality in PSNR between FGS and single-layer approaches with the dynamic test condition as defined in the MPEG document m8002 [12].

burst error with a loss of the first few frames in every GOV. Two burst lengths of one frame and seven frames are used for simulation. By applying the RFGS method for both the enhancement and base layers, the error drift is more serious as compared the drift for the RFGS1. However, the visual quality can still be quickly recovered from the burst errors.

We also perform the dynamic test following the channel bandwidth variation pattern as defined in [12] to demonstrate the performance of RFGS. The bandwidth pattern as illustrated in Fig. 2 are as follows. The total bandwidth is switched in a step size of 256 kbps that decreases from 1024 to 256 kbps and increases back to 1024 kbps. The instantaneous bit rate is held for 24 s (or 720 frames with frame rate of 30 fps). Other test conditions are identical to those described in Section V-A and as defined in [10]. In the simulation, the Novel sequence in CIF format and with the frame rate of 30 fps were used. The first 5040 frames of the sequence are used for testing and the base layer is coded at 256 kbps. During transmission, we use the Two-Level Priority Network, where the FGS base layer is set at high priority. When the bandwidth is small, the base layer will be sent first. For the single-layer approach, we encode the bitstream with 256, 512, 768, and 1024 kbps, and dynamically select the appropriate bitstreams for the target bit rates as defined in [12].

Fig. 18 shows the simulation results. As compared with the results based on the single layer and the baseline FGS approaches, the results show that the RFGS2 with the linear model can adaptively select the suitable α offline to achieve similar performance as that of the single-layer approach for given dynamic bandwidths and different scene over a long sequence.

As for the error-recovery speed for different sequences, shown in Fig. 17, it is observed that the error recovery is also related to the temporal dependency between the successive frames of the same sequence. For the fast-moving sequences like Coastguard and Foreman, the current frame only refers to a fraction of information from the reference frame, which allows limited error propagation. Thus, the errors vanish even with a larger leak factor α. For the slow-motion sequences such as Akiyo, most of the frames consist of static areas such that there exist strong dependencies between the consecutive frames of the sequence. The dependencies can improve the coding efficiency but suffer from more drift when the transmission bandwidth is insufficient. Therefore, the RFGS with a small α (about 0.5) is recommended for the slow motion video sequences to improve the error robustness.

VI. Conclusions

In this paper, we proposed a novel FGS coding technique, RFGS. The RFGS is a flexible framework that incorporates the ideas of leaky and partial predictions. Both techniques are used to provide fast error recovery when part of the bitstream is not available. The RFGS provides tools to achieve a balance between coding efficiency, error robustness, and bandwidth adaptation. The RFGS covers several well-know techniques such as MPEG-4 FGS, PFGS, and MC-FGS as special cases. Because the RFGS uses a high-quality reference, it can achieve improved coding efficiency. The adaptive selection of bitplane number can be used to allow the tradeoff between coding efficiency and error robustness. The coding efficiency is maximized for a range of the target channel bandwidth. The enhancement-layer information is scaled by a leak factor α, where $0 \leq \alpha \leq 1$ before adding to the base-layer image to form the high quality reference frame. Such a leak factor is also used to alleviate the error drift.

Our experimental results show that the RFGS framework can improve the coding efficiency up to 4 dB over the MPEG-4 FGS scheme in terms of average PSNR. The error recovery capability of RFGS is verified by dropping the first few frames of a GOV at the enhancement layer. It is also demonstrated that tradeoff between coding efficiency and error attenuation can be controlled by the leak factor α. We also provide an approach to

select the parameters and its performance approaches that of a near-optimal exhaustive search of parameters. Such a technique provides a good balance between coding efficiency and error resilience.

ACKNOWLEDGMENT

The authors wish to thank the anonymous reviewers for their insightful comments to improve the initial draft of this paper. The authors also wish to thank Dr. F. Wu for providing the original MPEG test sequences for the dynamic tests.

REFERENCES

[1] *Streaming Video Profile—Final Draft Amendment (FDAM 4)*, MPEG01/N3904.

[2] W. Li, "Overview of fine granularity scalability in MPEG-4 video standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, pp. 301–317, Mar. 2001.

[3] *Information Technology—Coding of Audio-Visual Objects Part 2: Visual ISO/IEC 14 496-2: 2001*, MPEG Video Group, ISO/IEC JTC1/SC 29/WG 11 N4350, July 2001.

[4] F. Wu, S. Li, and Y. Q. Zhang, "A framework for efficient progressive fine granularity scalable video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, pp. 332–344, Mar. 2001.

[5] M. Schaar and H. Radha, "Motion-compensation based fine-granular scalability (MC-FGS)," ISO/IEC JTC1/SC29/WG11, MPEG00/M6475, Oct. 2000.

[6] K. Y. Chang and R. W. Donaldson, "Analysis, optimization, and sensitivity study of differential PCM systems operating on a noisy communication channels," *IEEE Trans. Commun.*, vol. COM-20, pp. 338–350, June 1972.

[7] M. Ghanbari and V. Seferidis, "Efficient H.261-based two-layer video codecs for ATM networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 5, pp. 171–175, Apr. 1995.

[8] A. Fuldseth and T. A. Ramstad, "Robust subband video coding with leaky prediction," in *Proc. DSP Workshop*, Loen, Norway, Sept. 1996, pp. 57–60.

[9] S. Li, F. Wu, and Y.-Q. Zhang, "Experiment results with fine granularity scalable (PFGS) coding," ISO/IEC JTC1/SC29/WG11, MPEG99/M5742, Oct. 1999.

[10] *FGS Dxperiments*, ISO/IEC JTC1/SC29/WG11, MPEG00/N3316, Mar. 2000.

[11] F. Wu, S. Li, X. Y. Sun, and Y.-Q. Zhang, "Macroblock-based progressive fine granularity scalable coding," ISO/IEC JTC1/SC29/WG11, MPEG01/M6779, Jan. 2001.

[12] *Report on MPEG-4 Visual Fine Granularity Scalability Tools Verification Test*, ISO/IEC JTC1/SC29/WG11, MPEG02/M8002, Jan. 2002.

Hsiang-Chun Huang was born in Hsinchu, Taiwan, R.O.C., in 1977. He received the B.S. degree in electronics engineering from National Chiao-Tung University (NCTU), Hsinchu, Taiwan, R.O.C., in 2000, where he is currently working toward the Ph.D. degree in the Institute of Electronics Engineering.

His research interest is in streaming video compression.

Chung-Neng Wang was born in PingTung, Taiwan, R.O.C., in 1972. He received the B.S. degree in computer engineering from the National Chiao-Tung University (NCTU), HsinChu, Taiwan, R.O.C., in 1994, where he is currently working toward the Ph.D. degree in the Institute of Computer Science and Information Engineering.

His research interests are video/image compression, motion estimation, video transcoding, and streaming.

Tihao Chiang (SM'99) was born in Cha-Yi, Taiwan, R.O.C., in 1965. He received the B.S. degree in electrical engineering from the National Taiwan University, Taipei, Taiwan, R.O.C., in 1987, and the M.S. and Ph.D. degrees in electrical engineering from Columbia University, New York, in 1991 and 1995, respectively.

In 1995, he joined David Sarnoff Research Center, Princeton, NJ, as a Member of Technical Staff, and was later promoted to Technology Leader and a Program manager at Sarnoff. While at Sarnoff, he led a team of researchers and developed an optimized MPEG-2 software encoder. Since 1992, he has actively participated in ISO's MPEG digital video-coding standardization process, with particular focus on the scalability/compatibility issue. In September 1999, he joined the faculty at National Chiao-Tung University, Taiwan, R.O.C. He is currently the co-editor of part 7 of the MPEG-4 committee, and over the past ten years, has made more than 50 contributions to the MPEG committee. He holds 9 U.S. patents and 26 European and worldwide patents, and has published over 30 technical journal and conference papers in the field of video and signal processing. His main research interests are compatible/scalable video compression, stereoscopic video coding, and motion estimation.

Dr. Chiang was a co-recipient of the 2001 Best Paper Award from the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY. He received two Sarnoff Achievement Awards and three Sarnoff Team Awards for his work in the encoder and MPEG-4 areas.