PGOP: An Error Resilient Technique For Low Bit Rate and Low Latency Video Communications

Liang Cheng and Magda El Zarki
Donald Bren School of Information and Computer Science
University of California, Irvine, CA 92697
{lcheng61, magda}@ics.uci.edu

ABSTRACT

The provision of real-time low-latency video applications (e.g., interactive video tele-conferencing) over a wireless network is a significant research challenge. In this paper, we propose an encoder side error-resilient scheme (PGOP), by which we can stop the temporal error propagation in a progressive manner such that the constraint of encoder buffer delay can be satisfied. We compare the refreshing capability of our scheme to the well-known group of picture (GOP) scheme: with an equivalent refreshing capability, the PGOP scheme outperforms the GOP scheme in the low-latency scenario, showing better spatial and temporal quality for both frame-level and macroblock-level rate control modes. Furthermore, PGOP shows a significant decoding quality gain over GOP in a simulated wideband code division multiple access mobile network.

1 INTRODUCTION

The fundamental problem of generic wireless video communications is the ability of compressed video data to survive the lossy-natured mobile/wireless network. At the compression layer, the video stream is inevitably affected by residual transmission errors. The prediction loop at the compression layer, which is originally well devised to increase the compression efficiency, may spread errors temporally. The conventional scheme to safely stop the temporal error propagation is to insert an intra-coded frame (I-frame) for a specified interval. The I-frame does not use temporal prediction techniques and thus it is self-contained. Hence, the I-frame is considered as the key frame, providing both the function of "fidelity refreshing sync point" in the compression domain and the function of "random access" for play-back.

However, encoding an I-frame needs many more bits than a regular P-frame, for the same encoding quality. The excessive bits, when being transmitted through a fixed-rate channel, may accumulate in the encoder buffer and increase the buffer delay (i.e., $\geq$ 3s). For low-latency interactive video scenarios, the encoder delay is restricted (i.e., $\leq$ 0.5s), which makes it difficult for the encoder to regulate the bit rate without quality degradation. For the non-conversational real-time applications, such as video streaming, the concern with the buffer delay is replaced by the consideration about the cost and the area of the on-board buffering memories. This, again, emphasizes the importance of the proper buffer regulation.

The encoder side rate control is normally categorized into two types: the frame-level rate control and the macroblock-level (MB-level) rate control. When being applied to the GOP, these two control modes may produce different types of quality degradation. The frame-based rate control can preserve the spatial quality of the I-frame but may have to skip several subsequent P-frames to compensate for the excessive bits encoded in the I-frame. The frame skipping is subjectively perceived as "motion jerkiness". The MB-based rate control aims to avoid frame skipping by reducing the quality of the I-frame and thus the number of bits needed for encoding. This is characterized as spatial quality unevenness. Hence, when using I-frames in low-bandwidth low-latency scenarios, we fall into the dilemma of either "motion jerkiness" or "uneven" spatial quality. A variety of schemes have been proposed to solve this GOP problem [1] [2]. For example, in [1], the authors propose a frame-level rate control mechanism, by which they gracefully adjust the frame encoding rate in order to preserve the spatial quality and avoid the sudden frame skipping. In [2], the targeted number of bits per frame is allocated based on the distance of each P-frame to its corresponding I-frame in the same GOP such that they can decrease the frame skipping. However, these GOP adapted schemes cannot simultaneously improve both spatial and temporal quality at very low bit rates.

In a previous paper [3], we proposed an error resilient scheme: the progressive group of picture (PGOP), by which we can safely stop the temporal error propagation in a controllable and systematic manner. In that paper, we compare our scheme to other progressive I-frame techniques, such as adaptive intra refreshing (AIR), random intra-MB selection, and show that PGOP have superior performance under the same channel error conditions.

PGOP features progressive intra/inter mode selection. Like the GOP structure, PGOP is an encoder side error resilient tool and doesn't need any decoder implementation information. Hence, PGOP is suitable for heterogeneous decoding environments, e.g., the multicast video scenarioes. In addtion, considering the difficulty of obtaining instant channel packet loss information, PGOP is designed to be a preventive scheme. Nevertheless, we may use some *a priori* knowledge of the network to empirically adjust the refreshing capability of PGOP similar to what GOP does, i.e., adjusting the I-frame interval. In this paper, we continue this work by proving that PGOP is possessing an equivalent error refreshing capability to that of GOP. Based on this, we show that PGOP eases the encoding buffer regulation with less spatial/temporal quality degradation when compared to the GOP scheme. We propose that PGOP be used as a replacement for GOP in the low bit rate and low latency interactive wireless video applications.

The rest of the paper is arranged as follows. In Section 2, we briefly summarize the PGOP scheme. In Section 3, we analyze the refreshing capability of PGOP and compare it to that of GOP. In Section 4, we demonstrate that, given the same refreshing capability, PGOP solves the GOP problems mentioned earlier, i.e., the temporal quality unevenness and the spatial quality degradation in the low bit-rate low-latency scenario. Section 5 is the conclusion.

2 PROGRESSIVE GROUP OF PICTURE

In Figure. 1, we compare three heuristic intra-MBs allocation schemes: GOP based, row-by-row based and column-by-column based. We realize that if we allocate the intra-MBs on a column-by-column basis, the high-cost intra-MBs can be more evenly distributed than the other two schemes on the encoding scan order. Our PGOP is thus based on the heuristic column-by-column refreshing scheme.

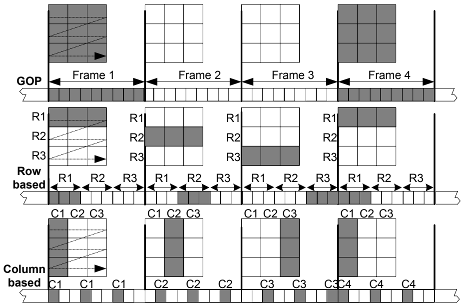
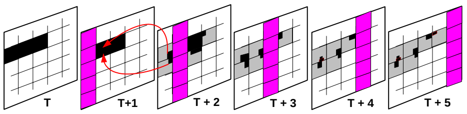
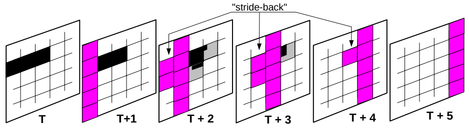


Figure 1: Intra-updating distribution of different refreshing schemes. The encoding modes are aligned based on the scanning (i.e., coding) order. The gray area denotes the intra-MBs; the white area denotes the inter MBs. R1, R2 and R3 denotes the 1^{st}, 2^{nd} and 3^{rd} row, respectively; C1, C2 and C3 denotes the 1^{st}, 2^{nd} and 3^{rd} row, respectively.



(a) Heuristic column refresh



(b) Error propagation adapted refreshing with *stride back*

☐ Unaffected MB
■ Lost MB/ Affected pixels
▨ MB having affected pixels
🟪 Refreshed MB

Figure 2: Illustration of the column-based refreshing and *stride back*.

However, the simple column-based refreshing scheme cannot completely stop the error propagation. As is illustrated in Figure 2-a, at time $(T+2)$, two MBs in the first column that are refreshed at time $(T+1)$ use the unrefreshed area for motion prediction. Even though this referenced unrefreshed area is being updated at time $(T+2)$, the existing errors at time $(T+1)$ will propagate to these two MBs. To solve this problem, we analyze the motion vectors (which give an indication of the error propagation) and augment the regular column-by-column refresh process by intra-updating the MBs affected by the residual error propagation. We call this *stride back*, as is seen in Figure 2-b. We are now able to prevent the error from spreading back across the refreshing column and the "dirty" area (where the error might exist) can be progressively reduced until the whole frame is completely refreshed.

We define the duration between the refreshing of the first MB-column and the refreshing of the last MB-column as a *basic PGOP*, M_b. Eq. (1) is used to calculate the *basic PGOP* interval.

$$M_b = \lceil \frac{N_c \times T_I}{N_R} \rceil, \tag{1}$$

where T_I is the interval between two frames with column refreshing; N_R is the number of columns being regularly refreshed in one frame; N_c denotes the total number of MB-columns in one frame (e.g., for QCIF,

$N_c = 11$). Note that the algorithm for adjusting the length of the *basic PGOP* is described in more detail in [3].

Upon the complete reception of a *basic PGOP*, the decoder can be guaranteed of a whole frame refresh. That is, errors that might exist in the motion compensation loop can be completely "refreshed" within the *basic PGOP*. Hence, if the channel loss happens when the $i^{th}(0 \leq i \leq N_c)$ column is being refreshed, the refreshing time (denoted as M) is bounded by M_u: $M \leq M_u$. Eq. (2) shows that M_u consists of the time for refreshing the rest of the frame $((N_c - i) \times \frac{T_I}{N_R})$ in the current *basic PGOP* plus the next *basic PGOP*.

$$M_u = \lceil \frac{(2 \times N_c - i) \times T_I}{N_R} \rceil \tag{2}$$

Thus, when considering the worst case of Eq. (2), i.e., $i = 0$, we have the absolute upper bound for M,

$$M \leq 2M_b \tag{3}$$

Eq. (3) shows that PGOP is able to recover the compressed stream from the frame loss within a maximum bound.

In practice, we may need less time than $2M_b$ to recover from a frame loss. We evaluate the realistic refreshing time of PGOP by using the video sequences, *Foreman*(400 frames, QCIF), *Carphone*(380 frames) and *Mother and Daughter*(300 frames). These sequences represent a typical real-time interactive video communication scenario: *Foreman* is considered to be the most motion intensive because it includes gesturing, camera shaking and camera panning; *Mother and Daughter* only consists of the gesturing and thereby represents the most static scenario; the motion intensity of *Carphone* is in between that of *Foreman* and *Mother and Daughter* because it has gesturing motion and camera shaking but no camera panning. For $T_I = 1$, $N_R = 1$ and $N_c = 11$, we evaluate the time that is needed to recover from every possible loss by iterating the frame loss and the decoding process for each frame. In Figure 3, we compare the frequency of the recovery time for these three sequences. We notice that in spite of the differences of these three video sequences, for most of the frames, when lost, we can recover the stream from the impact of the loss in no later than 13 frames, which is far less than the calculated absolute upper bound, i.e., 22 frames according to Eq. (1) and Eq. (3). Henceforth, we denote $\overline{M}$ as the average refreshing time, e.g., $\overline{M} = 13$ for these three sequences with the parameters: $T_I = 1$, $N_R = 1$ and $N_c = 11$.

3 REFRESHING CAPABILITY COMPARISON

In this section, we compare the refresh capability of GOP and PGOP under different lossy environments. We assume the minimal loss unit is one frame, which is the worst case for both GOP and PGOP. We use the frame loss rate (θ_l) to characterize the error situation of the network. First, we define a criterion to characterize the refreshing capability. We then compare PGOP and GOP by using this criterion.

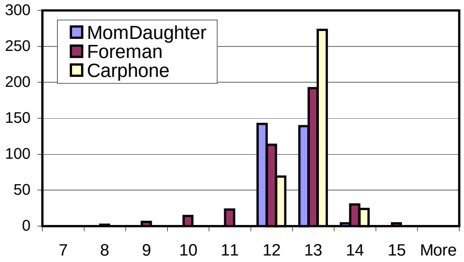


Figure 3: Refresh Time for the Loss of Each Frame

3.1 Refresh capability criterion

We notice that the essence of the error refreshing capability for both the PGOP and the GOP scheme is to stop the error propagation by eliminating the inter-frame dependency. For GOP, this is done by using the I-frame. Each lost frame may affect all of its subsequent frames until the next I-frame or the refreshing process is interrupted by another frame loss (Figure 4). For PGOP, this is done over an interval using a number of P-frames, thus the affected area by one lost frame is progressively reduced until the whole image is refreshed (Figure 5) or the refreshing is interrupted by another frame loss. We use Δ to denote the number of *error frames*, including the lost frame and all the following affected frames (or affected portion of frames in the case of PGOP) before the next lost frame. We use the frame affected rate (θ_a), i.e., the average number of *error frames* per unit of time or the frame refreshed rate $(1 - \theta_a)$, to evaluate the refreshing capability of these two schemes. Obviously, given the same θ_l, the higher θ_a means more frames (or portion of frames) are affected, or less frames (or portion of frames) are refreshed. The average number of error frames per lost frame is denoted by $E(\Delta)$, particularly $E(\Delta_{gop})$ for GOP and $E(\Delta_{pgop})$ for PGOP. Eq. (4) shows the relation among θ_a, θ_l and $E(\Delta)$.

$$\theta_a = \theta_l \times E(\Delta) \tag{4}$$

There is no denying that θ_a cannot directly represent the decoded video quality, which is generally considered as the ultimate goal for applying error resilience tools. However, as we mentioned in Section 1, the decoded video quality inevitably involves the decoder's

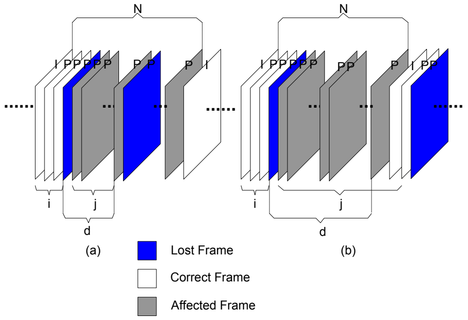


Figure 4: The affected frames by the lost frame in GOP

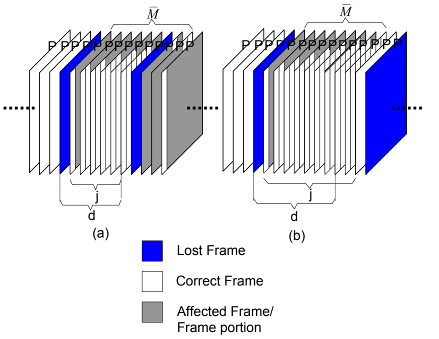


Figure 5: Affected frame/frame portion by the lost frame in PGOP

error concealment capability and thus it cannot accurately reflect the error refreshing capability provided by the encoder. In the following discussion, we will adjust the GOP and the PGOP to achieve equal $E(\Delta)$, by which we may have equal frame affected rate under the same frame loss rate (Eq. (4)). The tuning knob is the interval between I-frames for GOP and the duration of the *basic PGOP* interval for PGOP. We then discuss the encoding quality and the decoding quality of these two schemes.

3.2 Refresh capability of GOP and PGOP

In one GOP, we have one I-frame and a number of P-frames, i.e., *IPP..P*. We ignore B-frames because B-frames are not referenced by other types of frames. The number of frames in one GOP is defined as N. The loss in the motion compensation loop will not be refreshed until the next I-frame or the refresh process is interrupted by the next frame loss. In the worst situation, if an I-frame is lost, all of the P-frames in that GOP will be lost. (Note that we assume that no intra-MBs exist in P-frames.) We also assume that the probability of loss on each frame position in the GOP structure is equal, i.e., $\frac{1}{N}$. Now we denote the distance from the first frame (i.e. the I-frame) of the GOP structure to the lost frame by i, $0 \leq i \leq N-1$. The distance from the lost frame to the frame immediately before the next lost frame is j. j ranges from 1 (i.e., the next lost frame is immediately after the current lost frame) to $+\infty$. Recall that we use Δ_{GOP} to represent all of the frames affected by the lost frame in the GOP scheme. In Figure 4-a, we realize that Δ_{GOP} increases as the distance between two lost frames increases when two lost frames are within one GOP (i.e., $j < (N-i)$). When an I-frame is between two lost frames (i.e., $j \geq (N-i)$), Δ_{GOP} doesn't increase and it becomes a constant value (i.e., $(N-i)$) as is shown in Figure 4-b. The expected value of Δ_{GOP} is given by Eq. (5).

$$E(\Delta_{GOP}) = \sum_{i=0}^{N-1}\frac{1}{N}\times[\sum_{j=0}^{N-i-1}\theta_l\times(1-\theta_l)^j\times(j+1) + \sum_{j=N-i}^{+\infty}\theta_l\times(1-\theta_l)^j\times(N-i)] \quad (5)$$

Eq. (5) can be simplified to Eq. (6)

$$E(\Delta_{GOP}) = \frac{1}{\theta_l} - \frac{1-\theta_l}{N\times{\theta_l}^2}\times[1-(1-\theta_l)^N] \quad (6)$$

For the PGOP scheme, we progressively and systematically recover the stream from errors. We assume that the errors introduced to the motion compensation loop by the loss of different frames can be completely refreshed upon the reception of the next $\overline{M}$ frames. We denote the distance between two lost frames as j. The expected number of the *error frame* between two lost frames is $E(\Delta_{pgop})$. Figure 5-a shows that the PGOP refreshing process is interrupted by the next frame loss; Figure 5-b shows that the decoding process is completely recovered from the impact of the frame loss before the next frame loss. Similar to Eq. (5), we have Eq. (7) to describe the average number of *error frames* that are affected by a lost frame.

$$E(\Delta_{pgop}) = \sum_{j=0}^{\overline{M}-2}\theta_l\times(1-\theta_l)^j\times\sum_{i=0}^{j}\frac{\overline{M}-i}{\overline{M}} + \sum_{j=\overline{M}-1}^{+\infty}\theta_l\times(1-\theta_l)^j\times\sum_{i=0}^{\overline{M}-2}\frac{\overline{M}-i}{\overline{M}} \quad (7)$$

Eq. (7) can also be simplified to Eq. (6). That is, PGOP and GOP have an equivalent error refreshing capability for the same frame loss rate, as long as the I-frame interval of the GOP scheme equals the average refreshing time (i.e., $\overline{M}$) of PGOP. This is described in Eq. (8).

$$E(\Delta_{pgop}) = E(\Delta_{gop}) \ \leftarrow \ (N = \overline{M}) \tag{8}$$

4 PERFORMANCE

Numerous experiments have been conducted to evaluate the performance of the PGOP scheme. For the PGOP scheme, we set T_I to 1, N_R to 1 and N_c to 11. Thus the duration of the *basic PGOP* interval is 11(Eq. (1)). We choose a moderate value 13 for $\overline{M}$, which was shown in Section 2 to be suitable for most two-way video communication scenarios. By setting the I-frame interval of GOP to be 13, we can equalize the error refresh capability of these two schemes (Eq. (8) and Eq. (4)). In the following, we present the buffer-level variation, the encoding quality and the decoding quality. Note that for those frames that are skipped, the nearest previous frames are interpolated in order to calculate the impact of the frame skipping. We fix the following parameters in all of our experiments: the video format is QCIF; the encoding bit rate (i.e., R) is 64 kbps; the encoding frame rate (i.e., F) is 10 frames/s; the frame skipping margin is 80% of the encoding buffer size; the target buffer level is half the encoding buffer size.

Different rate control algorithms will result in different impacts on the encoding quality. We use the MPEG-4 [4] standardized rate control Q2 algorithm. In particular, we use frame-level Q2 algorithm for coarse rate control and the MB-level Q2 algorithm for fine rate control. Figure 6 shows the buffer-level variations and the encoding quality for both GOP and PGOP when using the MB-level rate control mode. We can see that: 1) the MB rate control effectively prevents the buffer from overflowing for both PGOP and GOP; 2) the encoding spatial quality degradation occurs every time an I-frame is coded for the GOP scheme; 3) with PGOP, the MB-level rate control is able to preserve a consistent spatial quality. It is notable that, in all of our experiments, the MB-level rate control can regulate the encoder bit rate such that frame skipping is hardly observed for both GOP and PGOP, even though we decrease the encoder buffer size to as low as 0.125s. Hence, we compare these two schemes based on the measurement of the spatial quality unevenness and the average quality. To characterize the spatial quality unevenness (i.e., Q_u), we calculate the quality difference of each frame and the frame immediately before it. We do it iteratively, and get the absolute difference value for each frame and then obtain the average, as is described in Eq. (9).

$$Q_u = [\sum_{i=1}^{N_S-1} (Q_i - Q_{i-1})]/N_S \tag{9}$$

where N_S denotes the number of frames in the sequence; Q_i denotes the PSNR quality of the i^{th} frame. The experimental results are reported in Table 1. It can be seen that, compared to the GOP scheme, the PGOP scheme significantly reduces the spatial quality unevenness and increases the average PSNR.

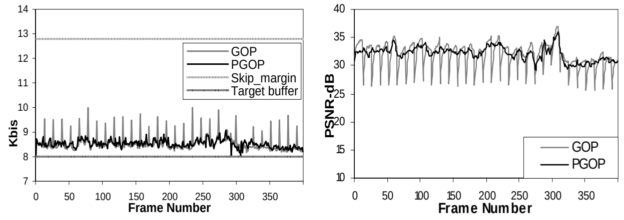
Table 1: Average quality and quality unevenness

	Q_u (dB)			Average PSNR (dB)		
	GOP	PGOP	Gain	GOP	PGOP	Gain
1	1.04	0.24	0.80	31.86	32.15	0.30
2	1.17	0.28	0.89	33.29	33.76	0.48
3	1.46	0.42	1.04	37.58	38.64	1.08

1: *Foreman*; 2: *Carphone*; 3: *MotherDaughter*

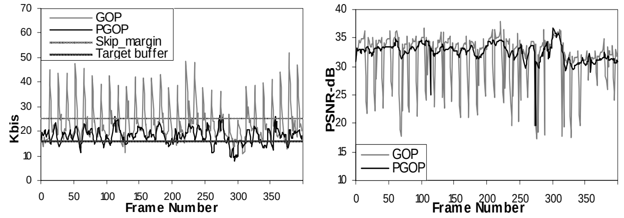
Figure. 7 shows the buffer variation and the encoding quality of GOP and PGOP with frame-level rate control. It can be seen that: 1) the buffer occupancy of the frame-level rate control shows more fluctuation than that of the MB-level rate control in general; 2) the buffer-level surge occurs periodically and thereby a considerable amount of frames are skipped with the GOP scheme; 3) the PGOP scheme facilitates the bit rate regulation and thus greatly reduces the fluctuation of the buffer level and the occurrence of buffer overflow. In Table 2, we show the experimental results for the frame-level rate control. It is noted that for the low encoder buffer delay configuration (i.e., Buffer Delay $\leq$ 0.5 s), the PGOP scheme outperforms the GOP scheme, skipping fewer frames, therefore having better motion continuity and achieving higher average encoding quality. To decrease the number of skipped frames for the GOP scheme, we have to increases the encoder buffer capacity, which results in longer buffer delay and is thereby unsuitable for real-time interactive video services.

To compare the error resilience of PGOP and GOP, we measure their decoding quality under six simulated WCDMA error patterns [5]. We repeat the video *Foreman* (400 frames) to form a 3 minutes long sequence and test the sequences with $\overline{M} = N = 13$. Note that the error concealment algorithm consists of simple MB replacement, i.e., we use the corresponding MBs in the previous frame to replace the lost MBs in the current frame. The results are tabulated in Table 3. We notice that the same decoder outputs better quality with PGOP than with GOP in all six WCDMA simulation environments. We believe that this difference primarily stems from the fact that progressive error refreshing provides more reliable temporal and spatial information around the lost data and thereby facilitates the decoder to conceal the lost data.



(a) Buffer-level variation (b) Encoding quality

Figure 6: MPEG-4 MB-based Q2 rate control. Buffer size = 16 kbits; Encoding buffer delay = 0.25s, sequence: *Foreman.*



(a) Buffer-level variation (b) Encoding quality

Figure 7: MPEG-4 frame-level Q2 rate control. Buffer size = 32 kbits, Encoding buffer delay = 0.5s, sequence: *Foreman.*

5 CONCLUSION

In this paper, we continue our past work by comparing the PGOP scheme to the GOP scheme. By adjusting the length of PGOP and GOP, we can equalize the refreshing capability of these two schemes based on the proposed criterion, i.e., frame affected rate. The experimental results show that PGOP is able to help the rate control mechanisms to meet the target bit rate more accurately, skip fewer frames, encode the sequences with higher visual quality, and maintain a lower fluctuation in the buffer occupancy under the constraint of low buffer delay. We also demonstrate that, with the fixed frame affected rate, PGOP outperforms GOP in terms of the decoding quality.

It is notable that PGOP is standard compliant and needs no coupling between the encoder and decoder, hence, it can facilitate any existing error control tools, such as FEC, ARQ, and decoder side error concealment algorithms.

References

[1] H. Song, J. Kim, C.-C. Jay Kuo, "Improved h.263+ rate control via variable frame rate adjustment and hybrid i-frame coding," *ICIP98*, 1998.

Table 2: Frame skipping and average encoding quality. ("Buf" is buffer delay in second)

Foreman

	Skipped Frame			PSNR (dB)		
Buf	GOP	PGOP	Gain	GOP	PGOP	Gain
0.25	137	19	118	30.67	32.31	1.64
0.5	101	3	98	31.54	32.50	0.96
1	64	1	63	32.18	32.50	0.31
2	20	0	20	32.81	32.52	-0.28
3	12	0	12	32.91	32.53	-0.38

Carphone

	Skipped Frame			PSNR (dB)		
Buf	GOP	PGOP	Gain	GOP	PGOP	Gain
0.25	124	10	114	33.37	34.21	0.85
0.5	107	1	106	33.61	34.21	0.61
1	48	1	47	34.12	34.20	0.08
2	24	0	24	34.49	34.21	-0.27
3	13	0	13	34.54	34.22	-0.32

Mother and Daughter

	Skipped Frame			PSNR (dB)		
Buf	GOP	PGOP	Gain	GOP	PGOP	Gain
0.25	106	27	79	38.63	39.56	0.93
0.5	95	14	81	38.82	39.59	0.78
1	90	17	73	38.84	39.64	0.80
2	73	0	73	39.17	39.58	0.41
3	62	0	62	39.18	39.60	0.41

Table 3: Decoding quality in WCDMA environments, Sequence: *Foreman.* MPEG-4 MB-level rate control; bit rate: 64 kbps; frame rate: 10; packet size: 640 bits. PGOP: $N_r = 1$, $T_I = 1$; GOP: $N = 13$.

Case	1	2	3	4	5	6
PGOP	29.01	31.52	29.59	31.37	29.71	31.48
GOP	27.96	31.21	28.85	31.13	28.93	31.28
Gain	1.05	0.31	0.74	0.24	0.78	0.20

[2] F. Pan, Z. Li, K. Lim and G. Feng, "A study of mpeg-4 rate control scheme and its improvements," *IEEE Trans. on Circuit and System for Video Tech.*, vol. 13, no. 5, May 2003.

[3] L. Cheng and M. El Zarki, "Perceptual quality feedback based progressive frame-level refreshing for robust video communication," *WCNC04*, Mar. 2004.

[4] "ISO/IEC 14496-2 (mpeg-4), Information technology - Coding of audio-visual objects Part 2: Video, international standard," 2002.

[5] Ericsson, Sweden, Standardization Sector of ITU, "WCDMA error patterns at 64 kb/s," *Proc. ITU-T Study Group 16. Multimedia Terminals and Systems Expert Group*, Jun. 1998.