Error-Resilience Transcoding Using Content-Aware Intra-Refresh Based on Profit Tracing

Chih-Ming Chen[1,2], Yung-Chang Chen[1]
[1]Department of Electrical Engineering
National Tsing Hua University
Hsinchu, Taiwan
ycchen@ee.nthu.edu.tw
[2]Chunghwa Telecom Co., Ltd.

Chia-Wen Lin
Department of Computer Science & Information Engineering
National Chung Cheng University
Chiayi, Taiwan
cwlin@cs.ccu.edu.tw

***Abstract*— In this paper, we present a two-pass error-resilience transcoding scheme based on content-aware intra-refresh (CAIR) for inserting error-resilience features to a compressed video. The proposed transcoder can adaptively vary the intra-refresh rate according to the video content and the channel's packet-loss rate to protect the most important macroblocks (MBs) against packet loss. Based on the CAIR transcoder, we propose a profit tracing scheme to improve the efficacy of intra-fresh allocation of the transcoder by avoiding wasting intra-refresh resource in MBs of high error-propagation ranks in a prediction path. Experimental results show that incorporating the proposed profit tracing scheme into CAIR scheme can achieve significant PSNR performance improvement over the CAIR scheme itself.**

I. INTRODUCTION

In a prestored video streaming system, a server stores pre-coded video streams and transmits them to client terminals for decoding and playback. There are several existing video coding techniques developed to compress video sequences into bit streams to reduce the data sizes. These video encoding techniques exploit spatial and temporal redundancy to achieve a high compression ratio, while making the compressed data very sensitive to transmission error. In the case of transmission error, the motion-compensated prediction (MCP) technique used in current standard video coders will result in drifting error due to the mismatch between the referenced frames used in the encoding and decoding, which usually leads to serious video quality degradation. Consider a three-tier streaming system using WLAN as an extension to the existing wired infrastructure, which offers local end-point devices the convenience of wireless connections. Video transport over WLANs has its own characteristics different from wired networks, such as high burst packet loss due to long and short fades, shadowing, and environmental noise. In order to achieve robustness for transmitting video over wireless networks, the server must be able to adapt the non-error-resilient compressed video streams into error-resilience-capable streams at the intermediate network node. To serve this purpose, a video transcoder [1][2] can be placed in a network node connected to a high-loss network to insert error-resilience features into the video bitstream to achieve robust video transmission over wireless channels [2-7].

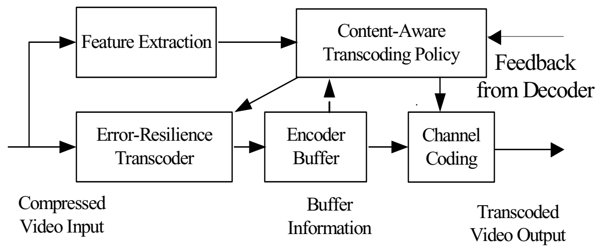


Fig. 1. Proposed system framework for error-resilience transcoding.

Fig. 1 illustrates the proposed error-resilience transcoder with feedback channels. The transcoder first extracts the video features (e.g., locations of video data which are likely to result in more serious error propagation if lost) from the incoming bitstream as well as estimates the client channel conditions according to the feedback channel statistics. The extracted features and the estimated channel condition are then used to determine the error-resilience coding policy. The features of video contents can be pre-computed in the front-end encoding process and sent to the transcoder as auxiliary data (metadata) to assist the transcoding. We have proposed in our previous work a two-pass error-resilience transcoding scheme by using Content-Aware Intra-Refresh (CAIR) [6][7]. In this work, we propose a profit tracing scheme to further improve the efficiency of intra-MB allocation in the CAIR method.

II. PROPOSED ERROR-RESILIENCE TRANSCODER

Fig. 2 shows the detailed structure of the proposed two-pass CAIR transcoder. At the first-pass front-end encoding of the proposed two-pass error-resilience transcoder architecture, the encoder utilizes the motion vectors and the estimated concealment distortion to estimate the error-propagation effect at the MB and frame levels within a group of pictures (GOP) as the side information which is stored in the streaming server to be used as hints to guide the error-resilient transcoding operation. In the second-pass transcoding process, the transcoder uses the side information received from the streaming server to determine the intra-refresh rate according to the channel statistics to determine the intra-refresh allocation for each frame of a GOP, and then performs intra-fresh on a number of high priority MBs with highest loss-impact factors based on the intra-refresh allocation. In the proposed scheme, most of the computation is done in the first-pass front-end encoding, which usually does not need to be done in real-time for prestored video applications. Only a small amount of computation is left to the second-pass transcoding, which usually has to meet the real-time requirement.

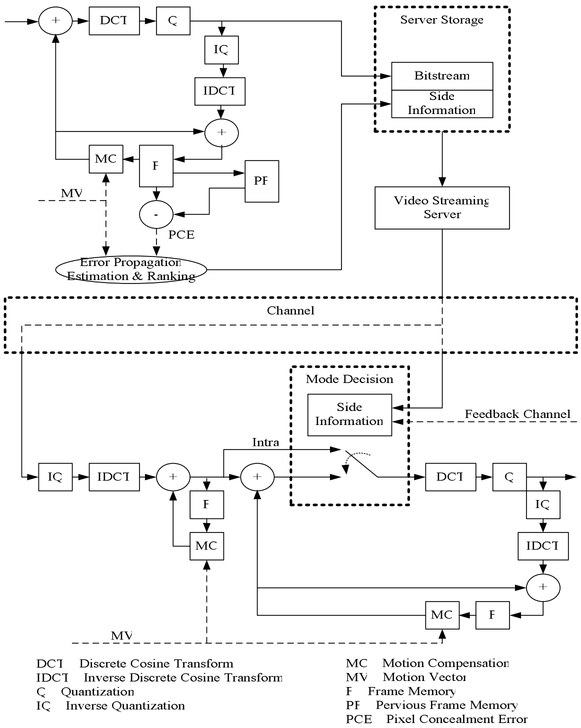


Fig. 2. Proposed two-pass error-resilience transcoder using content-aware intra-refresh.

III. ERROR RESILIENCE TRANSCODING USING CAIR

A. *Estimation of loss-impact*

To estimate the error propagation effect of a lost MB, we first define the pixel-level loss-impact (LI) metric as the product of two parameters: PRC (Pixel Reference Count) and PCE (Pixel Concealment Error), to characterize the amount of pixel-wise error propagation as follows:

$$LI(x,y,n) = PCE(x,y,n) \times PRC(x,y,n) \tag{1}$$

where $PRC(x,y,n)$ represents the frequency of pixel (x,y) of frame n being referenced by pixels in the following frames within a GOP in the motion-compensated prediction process. It can be calculated recursively by summing up the individual reference counts of pixels in frame $n+1$ which reference to pixel (x,y) of frame n in the reverse tracking order from the last frame to the first frame of a GOP as in (2). And $PCE(x,y,n)$, shown in (3), denotes the norm of concealment error of pixel (x,y) of frame n should this pixel be corrupted, where $f(x,y,n)$ is the pixel value of pixel (x,y) in frame n. In this work, the zero-motion error concealment scheme is adopted.

$$PRC(x,y,n) = \begin{cases} \sum\limits_{(x',y',n+1)\to(x,y,n)} PRC(x',y',n+1) & 1 \le n < N_{\text{GOP}} \\ 1 & n = N_{\text{GOP}} \end{cases} \tag{2}$$

$$PCE(x,y,n) = \left| f(x,y,n) - f(x,y,n-1) \right|^2 \tag{3}$$

We then use the motion information to calculate the current-frame's MB-level error-propagation (from the previous frames) as follows:

$$EP_{\text{MB}}(m,n) = \sum_{(x,y)\in \text{MB}_m} LI(x+MV_x, y+MV_y, n-1) \tag{4}$$

where m denotes the MB index in a frame; (x,y) denotes the pixel coordinate; n represents the time index; (MV_x,MV_y) represents the associated motion vector of pixel (x,y). Finally, all EP_{MB}'s in each frame are summed up to estimate the frame-level error-propagation as follows:

$$EP_n = \sum_{m=1}^{N_{\text{MB}}} EP_{\text{MB}}(m,n) \tag{5}$$

where N_{MB} denotes the number of MBs in a frame. After obtaining the above features in the first-pass front-end encoding, EP_{MB}'s of MBs and the frame-level EP_n are extracted and stored at the streaming server which will be sent to the intermediate transcoder as side information to enhance error resilience while streaming.

B. *Intra-Refresh Allocation of CAIR Transcoding*

In the transcoder, the CAIR scheme is used to determine the intra-refresh rate and the intra-MB allocation strategy for each GOP so as to adapt the transcoded video to varying network conditions. One key issue of the intra-refresh algorithm is to determine the number of MBs to be intra-coded in a GOP, which is determined using (6), where $N_{\text{intra}}^{\text{GOP}}$ is the total number of MBs to be intra-refreshed in a GOP; N_{GOP} is the GOP size; PLR is the channel packet loss rate which can be estimated via the client feedback information; TH_{intra} is a scaling parameter determined empirically [6].

$$N_{\text{intra}}^{\text{GOP}} = \frac{\dfrac{1}{N_{\text{GOP}}} \sum\limits_{n=1}^{N_{\text{GOP}}} EP_n \times PLR}{TH_{\text{intra}}} \tag{6}$$

The intra-refreshed MBs are then distributed to a GOP using the following algorithm,

if $n = 2$ (i.e., the first P-frame in a GOP)

$$N_{\text{intra}}^{n} = \frac{EP_n}{\sum\limits_{i=n}^{N_{\text{GOP}}} EP_i} \times N_{\text{intra}}^{\text{GOP}} \tag{7}$$

else if $3 \le n \le N_{\text{GOP}}$

$$N_{\text{intra}}^{n} = \frac{EP_n}{\sum\limits_{i=n}^{N_{\text{GOP}}} EP_i} \times \left(N_{\text{intra}}^{\text{GOP}} - \sum_{i=2}^{n-1} N_{\text{intra}}^{i} \right) \tag{8}$$

end if

where N_{intra}^{n} is the number of MBs to be intra-coded in frame n.

C. *CAIR with Profit Tracing*

In (4), the MB-level error-propagation EP_{MB} is estimated by summing up the pixel-level loss-impact values of the pervious frame that are referred by this MB in the MCP process. However, it has high possibility that two temporally correlated MBs in a prediction path between two successive frames both have high EP_{MB} ranks so as to be selected to be intra-refreshed for these frames. For example, as illustrated in Fig. 3, suppose that $MB(i+1,n)$ (MB_{i+1} of frame n) has a high EP_{MB} value: $EP_{\text{MB}}(i+1,n)$. The temporally correlated MBs of $MB(i+1,n)$ (e.g., $MB(i,n-1)$ and $MB(i+1,n-1)$ which are referenced partially by $MB(i+1,n)$) will also very likely have high EP_{MB} ranks since $EP_{\text{MB}}(i,n-1)$ and $EP_{\text{MB}}(i+1,n-1)$ inherit from $EP_{\text{MB}}(i+1,n)$ according to (4). Coding temporally correlated MBs in the intra mode will consume the intra-refresh budget rapidly, but may not be able to achieve comparable improvement on error resilience since the error propagation along the prediction path may have already been terminated by intra-refreshing an earlier MB in the path so that intra-refreshing its succeeding MBs is not that helpful. This will reduce the efficacy of the CAIR scheme.

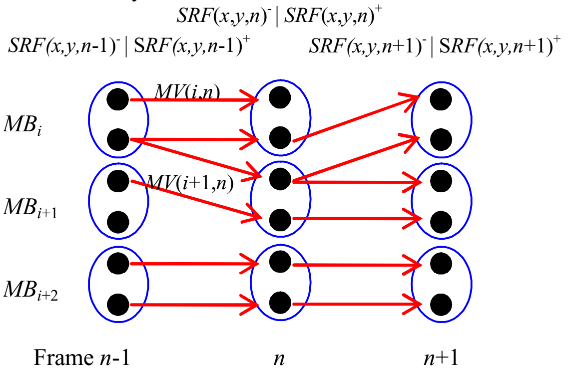


Fig. 3. Profit tracing of each refreshed MB.

To address the above problem of CAIR, we propose a profit tracing (PT) scheme to improve the intra-MB allocation strategy. First, we define a pixel-wise surplus refresh factor (*SRF*), which is propagated from a previous intra-coded MB. As shown in Fig. 3, $SRF(x,y,n)^-$ represents the intermediate *SRF* of pixel (x,y) of frame n before the transcoder decides the coding mode of the MB containing the pixel. $SRF(x,y,n)^-$ is inherited from the previous frame with motion vector (MV_x,MV_y), which is defined as the product of $SRF(x,y,n\text{-}1)^+$ and (1-*PLR*), as shown in (9).

$$SRF(x,y,n)^- = SRF(x+MV_X, y+MV_Y, n-1)^+ \times (1-PLR) \quad (9)$$

Since the loss probability of pixel (x,y) of frame n-1 is *PLR*, this pixel will has a probability (1-*PLR*) of providing a correct reference value to the succeeding MBs of frame n which reference to the pixel. Based on $SRF(x,y,n)^-$, the transcoder will determine the intra-MB allocation for frame n. After the mode decision, the intermediate value $SRF(x,y,n)^-$ will be transferred to a refreshed value $SRF(x,y,n)^+$ based on the coding mode. $SRF(x,y,n)^+$ is set to be 1, if pixel (x,y) belongs to an intra-refreshed MB. Otherwise, $SRF(x,y,n)^+$ remains the same as $SRF(x,y,n)^-$. The initial values, $SRF(x,y,0)^-$, are all set to 0. Besides, in an initial I-frame, the values of $SRF(x,y,0)^+$ are all equal to 1, as summarized in (10).

$$\begin{cases} SRF(x,y,n)^+ = 1, & (x,y)\in \text{intra-MB} \\ SRF(x,y,n)^+ = SRF(x,y,n)^-, & (x,y)\in \text{inter-MB} \\ SRF(x,y,0)^- = 0 & \\ SRF(x,y,0)^+ = 1 & \end{cases} \quad (10)$$

As depicted in Fig. 4, we use the motion information to map pixel-level $SRF(x,y,n\text{-}1)^+$ from the previous frame to obtain the MB-level $\overline{SRF}_{\mathrm{MB}}(m,n)$ as in (11), where $SRF(x,y,n)^-$ is calculated from $SRF(x,y,n\text{-}1)^+$ using (9) and $0 \le \overline{SRF}_{\mathrm{MB}}(m,n) \le 1$.

$$\overline{SRF}_{\mathrm{MB}}(m,n) = \frac{1}{Size_{\mathrm{MB}}} \sum_{(x,y)\in MB_m} SRF(x,y,n)^- \quad (11)$$

where $Size_{\mathrm{MB}}$ represnets the number of pixels in a MB.

After computing $\overline{SRF}_{MB}(m,n)$, we select a total of N^n_{intra} MBs with top-ranked $EP_{\mathrm{MB}}(m,n)\times\{1-\overline{SRF}_{\mathrm{MB}}(m,n)\}$ values to perform intra-refresh for the nth frame of a GOP.

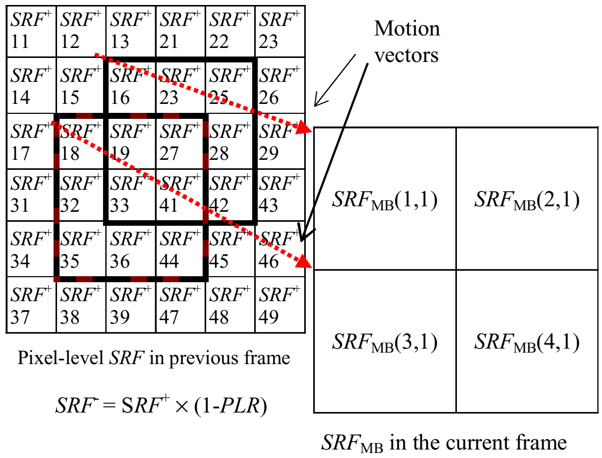


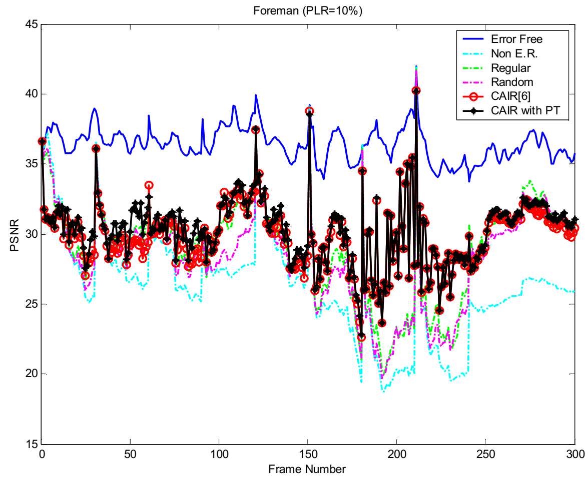
Fig. 4. Illustration of using MVs to map pixel-level SRF^+ of the previous frame to obtain MB-level SRF^- in the current frame.

IV. Experimental Results

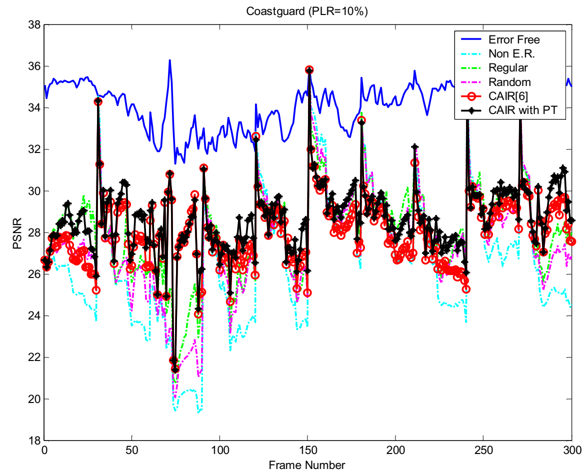
In our experiments, three QCIF (176×144) test sequences are pre-encoded using an MPEG-4 encoder at 30 fps and 384 Kbps. The group of picture (GOP) size is $(N_{\mathrm{GOP}}, M) = (30, 1)$, where M is the number of B-frames between two I/P-frames. In this work, we use a two-state Markov model to simulate a packet-erasure channel under four packet loss rates (PLR): 5%, 10%, 15%, and 20%, respectively. We adopt a simplified Gilbert channel at the packet level to generate ten packet-loss patterns for each PLR. The proposed method (CAIR with PT) is compared with our previously proposed CAIR method [6], the random intra-refresh [8], and the regular intra-refresh [8].

Table I.
Average PSNR values (in dB) of 10 packet-loss patterns using different intra-refresh transcoding schemes for four different PLRs

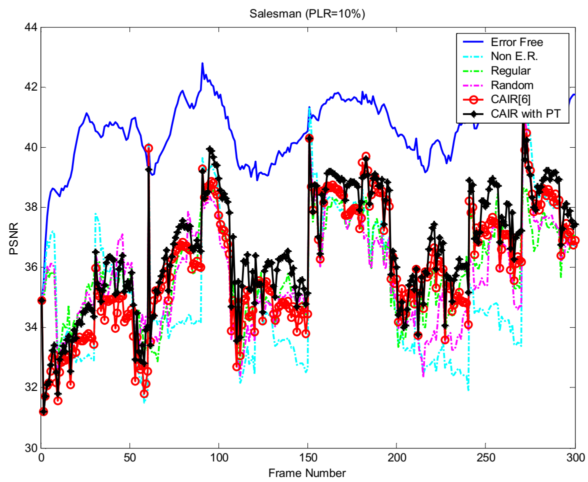
Sequence	Transcoding Method	Average PSNR for various PLRs			
		5%	10%	15%	20%
Foreman	Error Free	35.80			
	Non-E.R.	27.58	24.49	22.40	21.02
	CAIR with PT	**30.96**	**29.52**	**28.47**	**27.65**
	CAIR [6]	30.63	29.30	28.36	27.61
	Regular IR	28.78	27.05	25.99	25.28
	Random IR	28.41	26.51	25.93	25.37
Coastguard	Error Free	33.53			
	Non-E.R.	28.26	25.65	24.12	22.80
	CAIR with PT	**29.88**	**28.48**	**27.71**	**27.10**
	CAIR [6]	29.40	27.99	27.41	26.94
	Regular IR	29.02	27.56	26.94	26.16
	Random IR	28.72	27.07	26.78	26.21
Salesman	Error Free	39.81			
	Non-E.R.	36.84	34.32	32.24	30.60
	CAIR with PT	**37.33**	**36.29**	**35.39**	**35.40**
	CAIR [6]	37.14	35.90	35.17	34.48
	Regular IR	36.73	35.36	34.30	33.22
	Random IR	36.63	35.21	34.19	33.35



(a)



(b)



(c)

Fig. 5. Frame-by-frame PNSR performance comparison using various intra-refresh methods at PLR=10% for (a) *Foreman*; (b) *Coastguard*; (c) *Salesman*.

Fig. 5 depicts the frame-by-frame PNSR performance comparison of five different methods at *PLR*=10% for the three test sequences. Table I compares the average PSNR performances of the four methods computed from 10 loss patterns for each of the four PLRs. Experimental results show that the CAIR with profit tracing method can mitigate the error propagation due to packet loss more effectively than the other intra-refresh methods. For sequences of low motion activities it can improve PSNR by up to about 0.92 dB. Table I indicates that the proposed scheme achieves larger improvement on low-activity video than on high-activity ones. This is because the EP_{MB} value of a low-activity MB in a frame tends to be most inherited by a single MB rather than shared by several MBs in the following frame, thereby resulting in a longer sequence of high EP_{MB} MBs along a prediction path, which usually leads to poorer intra-refresh allocation efficiency in the CAIR transcoder. The results also show that the improvements under lower PLRs are typically larger than those under higher PLRs, because *SRF* will have a higher probability (1-*PLR*) to propagate to following frame in a low PLR situation.

V. Conclusion

We proposed a novel profit tracing scheme based on our previously proposed two-pass CAIR transcoder. The profit tracing scheme can improve the efficacy of intra-fresh allocation of the CAIR transcoder by avoiding wasting intra-refresh resource in MBs of high error-propagation ranks in the same prediction path. Experimental results show that the proposed transcoder can mitigate the error propagation due to packet-loss much more effectively so as to improve the visual quality significantly compared to regular and random intra-refresh transcoders. Incorporating the proposed profit tracing scheme into the CAIR transcoder can further achieve significant PSNR performance improvement over the CAIR scheme itself.

References

[1] J. Xin, C.-W. Lin, and M.-T. Sun, "Digital Video Transcoding," *Proc. IEEE*, vol. 93, no. 1, pp. 84-97, Jan. 2005.

[2] J. Xin, A. Vetro, and H. Sun, "Error resilience video transcoding for wireless communications," *IEEE Wireless Commun*, vol. 12, no. 4, pp. 14-21, Aug. 2005.

[3] G. de los Reyes, A. R. Reibman, S. F. Chang, and J. C. I. Chuang, "Error-resilient transcoding for video over wireless channels," *IEEE J. Select. Areas Commun.*, vol. 18, pp. 1063-1074, June 2000.

[4] S. Dogan, A. Cellatoglu, M. Uyguroglu, A. H. Sadka, and A. M. Kondoz, "Error-resilient video transcoding for robust internetwork communications using GPRS," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 6, pp. 453-464, June 2002.

[5] Z. He, J. Cai, and C.-W. Chen, "Joint source channel rate-distortion analysis for adaptive mode selection and rate control in wireless video coding," *IEEE Trans. Circuits Syst. Video Technol.* vol. 12, no. 6, pp. 511-523, June 2002.

[6] H.-J. Chiou, Y.-R. Lee, and C.-W. Lin, "Content-aware error resilient transcoding using prioritized intra-refresh for video streaming," *J. Visual Commun. Image Represent.*, vol. 16, no. 4-5, pp. 563-588, Aug.-Oct. 2005.

[7] C.-M. Chen, Y.-R. Lee, C.-W. Lin, and Y.-C. Chen "Adaptive error resilience transcoding for video multicast over wireless networks," in *Proc. IEEE Int. Conf. Multimedia and Expo*, July 2005, Amsterdam, the Netherlands.

[8] G. Cote and F. Kossentini, "Optimal intra coding of blocks for robust video communication over the internet," Signal Processing: Image Commun., Special Issue on Real-time Video over Internet, vol. 15, pp. 25-34, Sep. 1999.