Content-Aware Write Reduction Mechanism of Phase-Change RAM based Frame Store in H.264 Video Codec System

Sanchuan Guo, Zhenyu Liu, Guohong Li and Dongsheng Wang
Tsinghua National Laboratory for Information Science and Technology
Department of Computer Science and Technology, Tsinghua University
Beijing 100084, China
Email: {guosc07,ligh08}@mails.tsinghua.edu.cn, {liuzhenyu73,wds}@mail.tsinghua.edu.cn

***Abstract*— H.264 video codec system requires big capacity of Frame Store (FS) for buffering reference frames. The up-to-date Phase-change Random Access Memory (PRAM) is the promising approach for on-chip caching the reference signals, as PRAM offers the advantages in terms of high density and low leakage power. However, the write endurance problem, that is a PRAM cell can only tolerant limited number of write operations, becomes the main barrier in practical applications. This paper studies the wear reduction techniques of PRAM based FS in H.264 codec system. On the basis of rate-distortion theory, the content oriented selective writing mechanisms are proposed to reduce bit updates in the reference frame buffers. Experiments demonstrate that, for typical video sequences with different frame sizes, our methods averagely achieve more than 30% reduction of bit updates, while introducing around 20% BDBR cost. The power consumption is reduced by 55% on average, and the estimated PRAM lifetime is extended by 61%.**

I. Introduction

High definition video codec requires a high capacity of memory in the video codec system. In the most widely used H.264 Video Codec System, the off-chip storage which mainly used to store the reference frames, is named Frame Store (FS) and has a large capacity. Currently, FS is implemented by the off-chip DRAM. As the frame size get larger and Multiple Reference Frames (MRF) is adopted, the size of FS grow linearly with the video frame size and the number of reference frames. With the feature size of DRAM shrinks, the leakage power of DRAM increases. The high power consumption of DRAM restricts its practical application of high definition codec, especially on mobile devices, of which the size and the power budget are quite limited [1].

To overcome the obstacle of off-chip memory size as well as power consumption, Phase-change Random Access Memory (PRAM) is one possible solution. PRAM is one of the emerging non-volatile memories. As compared with DRAM, PRAM has increased the density by up to 300% [2]. Furthermore, due to the non-volatile nature, PRAM does not need refreshing, and its leakage power is very low. PRAM offers a comprehensive solution to the burgeoning size and power problems of the traditional DRAM storage.

However, the write endurance issue is the critical challenge of replacing DRAM with PRAM. A PRAM cell can only tolerate 10^8 - 10^9 write operations [3]. To push PRAM in the industrial applications, numerous efficient methods for PRAM write reduction have been suggested in literatures [3][4]. Differential write (DR) is one simple but effective scheme, in which the existing data and new data to be written are compared in a bit-by-bit manner, and only the different bits are updated in PRAM [4]. Even employing the differential writing strategy, the updates to PRAM cells are not distributed uniformly, and PRAM lifetime is limited by the most used cell. Wear leveling methods are proposed to distribute writes over PRAM cells in a uniform way [3]. The aforementioned proposals are developed aiming to the general-purpose applications. For video applications, Kwon et al. minimize the number of writes to PRAM by using the lossless compression methods [5].

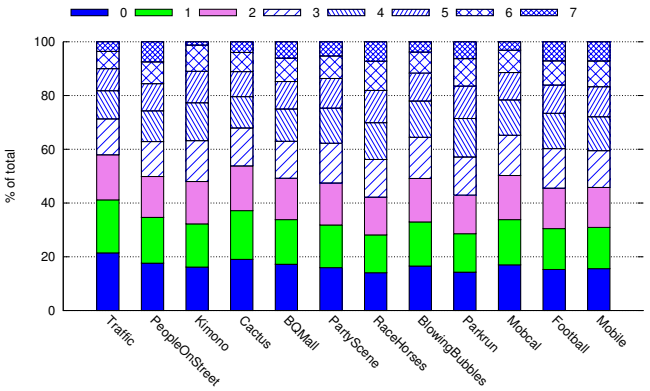

Fig. 1. Write density statistics of each bit in a pixel in reference frame buffer

In H.264 video coding, we notice that each bit of the reference pixels possesses different importance. Because of the temporal locality between neighboring frames and the spatial locality among the nearby pixels in the same picture, the pixels in successive frames with the same coordinate always have the approximate values. In consequence, the least significant bits (LSBs) updates more frequently than the most significant ones (MSBs). To verify this variation, we profiled the updates to each bit inside the reference frame pixels by using JM reference software. Figure 1 shows the statistics. It can be observed that, on average, the least 3 significant bits contributes more than 50% of the bit updates. In contrast, the update of the most 3 significant bits merely accounts for 26% on average. As the ith bit in one pixel has the weight 2^{i-1}, saving the write of LSBs leads to smaller coding quality degradation than MSBs.

In this paper, leveraging on the investigation of texture and motion features of current image block, we propose Content-

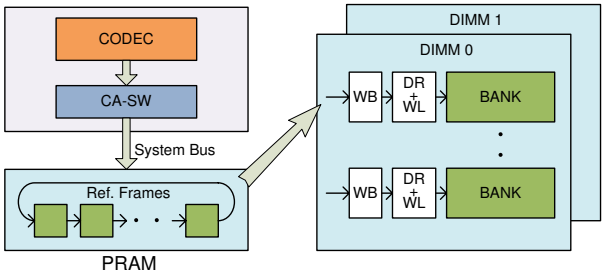

Fig. 2. System architecture using PRAM as the external FS memory (WB: write buffer; DR: bit-level differential writing; WL: wear-leveling; CA-SW: content-aware selective writing)

Aware Selective Writing mechanism to reduce the number of write operation to reference frame buffers. The proposed algorithm was embedded in the JM reference software to demonstrate its performance. It was verified that averagely more than 30% write operations to FS can be saved at the cost of the averaged 20% bit-rate increase. The power consumption is reduced by 55% on average, and the estimated PRAM lifetime is extended by 61%. Moreover, our lossy methods are orthogonal to the previously proposed lossless mechanisms.

II. Content-Aware Selective Writing Mechanism

In this research, we assume a H.264 coding chip with the PRAM based external FS memory, as shown by Fig. 2. The reference frame loop buffer is implemented by the PRAM. It is further assumed that bit-level differential writing and bit-level wear-leveling have been both used in the PRAM.

Our wear reduction design employs the Content-Aware Selective Writing to the reference frames in FS. Different from the original H.264 codec system, in our design, the pixels in the reconstructed frames are analyzed in terms of their importance, texture and motion features, and then adaptively written to the PRAM in bit-wise grain. It should be noticed that, to avoid the drifting, the encoder and decoder must adopt the same reference frame updating mechanism. The proposed methods contribute to the external memory traffic in both encoder and decoder sides. Of course, our selective writing leads to the addition noise in the reference signals, and consequently degrades the coding quality.

The principle of our method relies on dynamically omitting the update to one or more lowest bits in reference pixels, on the basis of the estimation of prediction residues. From the analysis of literature [6], the bit-wise write saving to the reference pixel can be by an additional white-noise source to the prediction residues. Let capital letter $S(u,v)$ represent the disctrete 2-D DCT transform coefficients of one 4×4-block prediction residues. Let Δ_{ee} denote the power spectral density of noise. With rate-distortion theory [7], when $S(u,v)$ is identified as memoryless signal, the distortion D and the corresponding rate R_D have the relations described by

$$\begin{cases} D = \min(\Theta, S_{ee}(u,v) + \Delta_{ee}) \\ R_D = \max(0, \frac{1}{2}\log_2 \frac{S_{ee}(u,v) + \Delta_{ee}}{\Theta}) \end{cases} \quad (1)$$

where, $S_{ee}(u,v)$ is the power spectral density of $S(u,v)$, Θ can be approximated as the quantization noise. Θ is linearly with the square of quantization interval (Q), i.e.,

$$\Theta = \frac{Q^2}{12}. \quad (2)$$

From (1), if it is assumed that $S_{ee}(u,v) \gg \Delta_{ee}$, with Taylor series, the augment of rate cost (dR_D) can be approximated as

$$dR_D = \frac{\Delta_{ee}}{S_{ee}(u,v)}. \quad (3)$$

Therefore, the adverse effect to the rate of reference pixel noise diminishes with the increase of DCT coefficients energy. From literature [6], the noise coming from saving the lowest l-bit in reference frames can be modeled as

$$\Delta_{ee} = \frac{\left(2^l\right)^2}{\gamma} \quad (4)$$

If the increased rate cost dR_D is desired to be equal to βR, from (1), (2), (3) and (4), we can deduce the value of l as

$$l = \frac{1}{2}\log_2\left(\beta \cdot \gamma \cdot S_{ee}\log_2\frac{12 \cdot S_{ee}}{Q^2}\right) \quad (5)$$

In our work, one important topic comes from the prediction of $S_{ee}(u,v)$. $S_{ee}(u,v)$ present the DCT coefficients using the current decoded image pixels as the predictions. In the stage of storing the current pixels in the PRAM, we can not derive the value of $S_{ee}(u,v)$. However, it is feasible to predict $S_{ee}(u,v)$ via the investigation of the texture and the motion feature of current image block. Let $e(i,j)$ ($i \in [0,3], j \in [0,3]$) denote the prediction residues. According to Parseval's theorem, we have

$$\sum_{u=0}^{3}\sum_{v=0}^{3} S_{ee}(u,v) = \sum_{i=0}^{3}\sum_{j=0}^{3} e(i,j)^2. \quad (6)$$

It is reasonable to expect the large value of $S_{ee}(u,v)$ with the increase of prediction residue power.

As mention in [8], we can derive the estimation of prediction residue power from the edge and motion characters. In this analysis, the impact of the edge intensity of the source image on the prediction errors is investigated in the spatial domain. In order to simplify the mathematical description, the analysis is first restricted to one-dimensional (1-D) spatial signals, as shown in Fig. 3, and the quantization noise is temporarily ignored. $s_t(x)$ and $s_{t-1}(x)$ denote the spatial-continuous signals at time instance t and $t-1$. $s_t(x)$ is a displaced version of $s_{t-1}(x)$ and the distance is d_x, which can be expressed as $s_t(x) = s_{t-1}(x - d_x)$. These continuous image signals are sampled by the sensor array before digital processing. The spatial sampling interval is denoted as u_x. The displacement estimation error is $\Delta_x = d_x - \text{round}(d_x/u_x) \cdot u_x$.

From Fig. 3, the prediction error $e(i \cdot u_x)$ of pixel i can be approximated as

$$e(i \cdot u_x) \approx \Delta_x \cdot s'_t(i \cdot u_x) \quad (7)$$

where, $s'_t(i \cdot u_x)$ is the edge gradient of $s_t(x)$ at the ith camera sensor and the displacement estimation error Δ_x is a random variable with zero mean and $\Delta_x \in [-u_x/2, u_x/2]$.

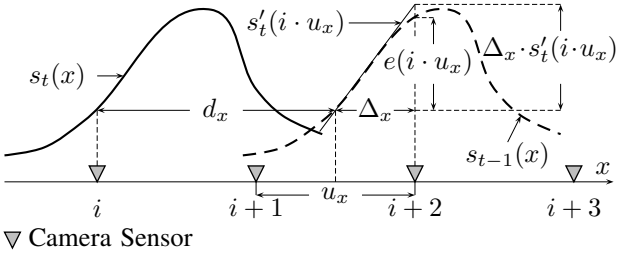

Fig. 3. Analysis of 1-D prediction error caused by edge gradient and displacement estimation error

From (7), we can see that the power of prediction residue is mainly determined by two factors. That is, when one image block possesses complex textures and motions, it has high probability to get the large values of DCT coefficients.

Based on the above analysis, we propose the following Content-Aware Selective Writing algorithm. The image content is estimated in 4×4-blocks. We first calculate the edge vectors within the 4×4-block by using 2×2 edge detection operator as follows.

$$\begin{cases} gx_{i,j} = p_{i+1,j} + p_{i+1,j+1} - p_{i,j} - p_{i,j+1} \\ gy_{i,j} = p_{i,j+1} + p_{i+1,j+1} - p_{i,j} - p_{i+1,j} \end{cases} \tag{8}$$

where $p_{i,j}$ $(i \in [0,3], j \in [0,3])$ denotes the picture pixel value, and $gx_{i,j}$ and $gy_{i,j}$ represent the edge gradient in horizontal and vertical directions.

Let mv_x and mv_y denote the motion vector of the current 4×4-block. The approximate value of $S_{ee}(u,v)$, i.e., $\widetilde{S}_{ee}$, is written as

$$\widetilde{S}_{ee} = \frac{1}{16} \sum_{i=0}^{3} \sum_{j=0}^{3} \left(gx_{i,j}^2 \left[\frac{\mathrm{mod}(\frac{mv_x}{4})}{4} \right]^2 + gy_{i,j}^2 \left[\frac{\mathrm{mod}(\frac{mv_y}{4})}{4} \right]^2 \right) \tag{9}$$

Then we can use (5) to calculate l. In (5), we let $a = \beta \cdot \gamma$ as an experimental parameter. The lowest l-bits of pixel values in the 4×4-block is not written to FS.

III. Experiment

In our experiments, we integrated our proposals into JM17.0 reference software and the original algorithm is used as the anchor. The simulation conditions were defined according to the recommendations in [9]. We used 12 typical video sequences with various frame sizes, and each sequence with 100 frames and quantization parameter values of 22, 27, 32, and 37 were tested. IPPP GOP and a single slice per picture were used for all sequences. The number of reference frames is set as 5, and Fast Motion Estimation (FME) is enabled for encoding speed. In the evaluations of the coding quality of proposed algorithms, BDBR (Bjonteggard Delta BitRate) and BDPSNR (Bjonteggard Delta PSNR) [10], which are respectively the average difference of bit-rate and PSNR between two methods, were applied to produce the quantitative analysis. For the experimental parameter a in (5), we have two sets of experiments with $a = 1$ and $a = 2$, to see the effect of different values of parameter a.

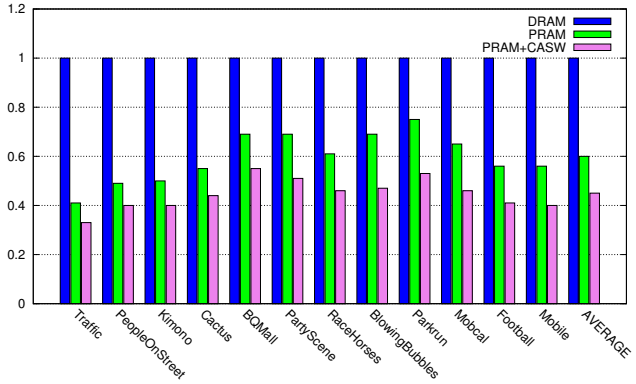

Fig. 4. Power consumption comparison of DRAM FS, PRAM FS and PRAM FS with Content-Aware Selective Writing mechanism. Normalized to DRAM FS. (QP=22; 100 frames; In CA-SW the parameter $a = 1$)

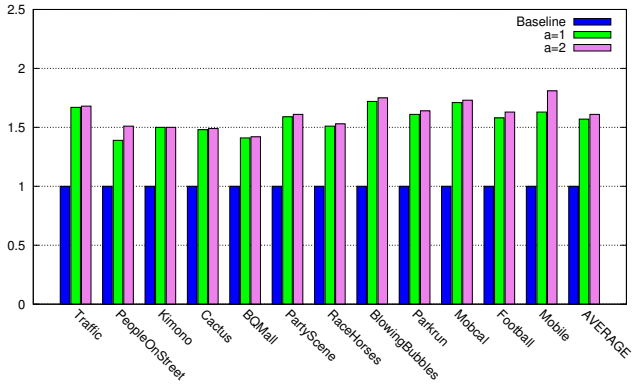

Fig. 5. Lifetime comparison of PRAM FS baseline and PRAM FS with Content-Aware Selective Writing mechanism, under the parameter $a = 1$ and $a = 2$. Normalized to PRAM FS baseline.

Table I illustrates the coding efficiency as well as the write reduction performance of our mechanism with $a = 1$ and $a = 2$. On average, when $a = 1$, the bit updates are reduced by 33.8% over the baseline system, which shows the mechanism is very effective in reducing writes to PRAM FS. The overhead is that the bit rate has a increase of 17.4%. For $a = 2$ configuration, the bit updates can be reduced by 35.5%, while the bit rate increase is 20.6% over the baseline. So we can see that with the increase of a by 1, BDPSNR has dropped by 0.1dB, and BDBR has increased by about 3% of the baseline. Also, 1.7% more write reduction is achieved.

To evaluate the power consumption of the PRAM FS design as well as our CA-SW algorithm, we calculated the power consumption of FS in three configurations, i.e. baseline DRAM FS, PRAM FS without write saving algorithm and PRAM FS with our CA-SW algorithm. The power parameters are obtained from CACTI for DRAM[11] and its revised version for PRAM[12]. In the coding procedure, QP is set equal to 22, and the parameter a is set to 1 in PRAM+CASW configuration. The power consumption statistics including both static and dynamic power are shown in Fig. 4. By replacing DRAM FS with PRAM, the total power consumption is reduced by 40% on average. Due to the non-volatile nature of PRAM, refreshing is not need, so the static power is much lower than

TABLE I

CODING EFFICIENCY AND WRITE REDUCTION OF CONTENT-AWARE SELECTIVE WRITING MECHANISM

Sequences	Frame Size	$a = 1$			$a = 2$		
		BDPSNR (dB)	BDBR (%)	Write Reduction (%)	BDPSNR (dB)	BDBR (%)	Write Reduction (%)
Traffic	2560x1600	-0.75326	22.906	35.3	-0.85947	27.122	36.2
PeopleOnStreet	2560x1600	-0.32295	7.5186	27.1	-0.68833	16.276	30.8
Kimono	1920x1080	-0.72805	22.254	28.9	-0.75886	23.822	29.2
Cactus	1920x1080	-0.42998	21.499	28.4	-0.47020	23.010	28.9
BQMall	832x480	-0.70315	16.853	26.5	-0.75597	18.451	27.0
PartyScene	832x480	-0.64755	13.563	34.3	-0.70343	15.123	35.4
RaceHorses	416x240	-0.81855	17.925	33.4	-0.89044	20.182	34.5
BlowingBubbles	416x240	-0.83660	22.217	41.1	-0.90330	24.800	42.2
Parkrun	1280x720	-0.53966	11.438	37.0	-0.59162	12.806	38.4
Mobcal	1280x720	-0.69459	28.608	38.5	-0.79121	33.647	40.0
Football	352x288	-0.38532	7.3701	36.5	-0.45032	8.6010	38.7
Mobile	352x288	-0.74791	16.972	38.8	-1.0243	23.030	44.6
Average		-0.63396	17.427	33.8	-0.74062	20.573	35.5

that of the DRAM counterpart. Although the dynamic power consumed by read and write access is higher for PRAM than DRAM, the total consumption is sharply reduced. For PRAM configuration employing the proposed CA-SW mechanism, 15% more power saving is achieved. This is due to the dynamic power reduction for write operations.

To evaluate the lifetime increase of CA-SW mechanism to PRAM, we use the lifetime model proposed in prior work [13]. For simplicity, we assume that the wear-leveling scheme achieves a perfectly even distribution of writes among all PRAM cells, so the lifetime of the whole PRAM (in seconds) is expressed as

$$L = \frac{w_{PRAM} \cdot N}{\sum_{i=0}^{N} w_i} \tag{10}$$

where w_{PRAM} is the number of writes allowed to a PRAM cell. We assumed $w = 10^8$, which are used for recent previous work [3][14]. N is the total number of PRAM cells and w_i is the writes per seconds of the i-th cell. In our experiment, we let the first 100 frames from a video sequence be encoded repeatedly, with QP=$\{22, 27, 32, 37\}$, in order to get the average lifetime across different quantization parameters. The lifetime of PRAM of the baseline and system with CA-SW is shown in Fig. 5. With the parameter $a = 1$, our algorithm achieves a lifetime extension of 57% on average over the baseline. With $a = 2$, more 4% lifetime improvement is achieved. We can say that with bit-level wear-leveling mechanisms, our CA-SW algorithm is effective in extending the lifetime of PRAM.

IV. CONCLUSION

As the high definition specifications become popular or even mandatory in the modern codec system design, Frame Store (FS) size and power consumption are the obstacles that the traditional off-chip DRAM faces. In this paper, we have studied using Phase-change RAM (PRAM) as the FS in H.264 video codec system, which offers high density and low leakage power. We propose Content-Aware Selective Writing mechanism to tackle the write endurance problem with PRAM. Based on rate-distortion theory, selective writing is performed to PRAM FS, according to the image content investigation. The experiment using JM reference software with 12 typical video sequences of different size shows that our CA-SW method averagely achieves more than 30% reduction of bit updates, while introducing about 20% BDBR cost. The power consumption is reduced by 55% on average, and the estimated PRAM lifetime is extended by 61%.

ACKNOWLEDGMENT

This work is supported by Nature Science Foundation of China under Grant No. 60833004, the National 863 High-Tech Program of China (No.2012AA010905), and TNList cross-discipline foundation.

REFERENCES

[1] Y. Xie, "Modeling, architecture, and applications for emerging memory technologies," *Design & Test of Computers, IEEE*, vol. 28, no. 1, pp. 44–51, 2011.

[2] M. Qureshi, V. Srinivasan, and J. Rivers, "Scalable high performance main memory system using phase-change memory technology," in *ACM SIGARCH Computer Architecture News*, vol. 37, no. 3. ACM, 2009, pp. 24–33.

[3] P. Zhou, B. Zhao, J. Yang, and Y. Zhang, "A durable and energy efficient main memory using phase change memory technology," in *ACM SIGARCH Computer Architecture News*, vol. 37, no. 3. ACM, 2009, pp. 14–23.

[4] B. Yang, J. Lee, J. Kim, J. Cho, S. Lee, and B. Yu, "A low power phase-change random access memory using a data-comparison write scheme," in *Circuits and Systems, 2007. ISCAS 2007. IEEE International Symposium on*. IEEE, 2007, pp. 3014–3017.

[5] S. Kwon, S. Yoo, S. Lee, and J. Park, "Optimizing video application design for phase-change ram-based main memory," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, no. 99, pp. 1–9.

[6] A. Oppenheim and C. Weinstein, "Effects of finite register length in digital filtering and the fast fourier transform," *Proceedings of the IEEE*, vol. 60, no. 8, pp. 957–976, 1972.

[7] T. Berger, *Rate Distortion Theory*. Prentice Hall, 1971.

[8] Z. Liu, J. Zhou, S. Goto, and T. Ikenaga, "Motion estimation optimization for h. 264/avc using source image edge features," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 19, no. 8, pp. 1095–1107, 2009.

[9] F. Bossen, "Common test conditions and software reference configurations," *JCTVC-B300, Geneva, Switzerland*, 2010.

[10] G. Bjontegard, "Calculation of average psnr differences between rd-curves," *ITU-T VCEG-M33*, 2001.

[11] (2008) The cacti website. [Online]. Available: http://www.hpl.hp.com/research/cacti/

[12] X. Dong, N. Jouppi, and Y. Xie, "Pcramsim: System-level performance, energy, and area modeling for phase-change ram," in *Proceedings of the 2009 International Conference on Computer-Aided Design*. ACM, 2009, pp. 269–275.

[13] Y. Joo, D. Niu, X. Dong, G. Sun, N. Chang, and Y. Xie, "Energy- and endurance-aware design of phase change memory caches," in *Proceedings of the Conference on Design, Automation and Test in Europe*. European Design and Automation Association, 2010, pp. 136–141.

[14] B. Lee, E. Ipek, O. Mutlu, and D. Burger, "Architecting phase change memory as a scalable dram alternative," in *ACM SIGARCH Computer Architecture News*, vol. 37, no. 3. ACM, 2009, pp. 2–13.