

 Open access • Journal Article • DOI:10.1109/TCSVT.2006.887130

Single Reference Frame Multiple Current Macroblocks Scheme for Multiple Reference Frame Motion Estimation in H.264/AVC — [Source link](#)

Tung-Chien Chen, Chuan-Yung Tsai, Yu-Wen Huang, Liang-Gee Chen

Institutions: National Taiwan University

Published on: 01 Feb 2007 - IEEE Transactions on Circuits and Systems for Video Technology (IEEE)

Topics: Reference frame, Residual frame, Inter frame, Frame (networking) and Memory bandwidth

Related papers:

- [On the data reuse and memory bandwidth analysis for full-search block-matching VLSI architecture](#)
- [Fast multiple reference frame motion estimation for H.264/AVC](#)
- [Fast multiple reference frame selection algorithm in H.264/AVC](#)
- [H.264/AVC multiple reference video frames](#)
- [Level C+ data reuse scheme for motion estimation with corresponding coding orders](#)

Share this paper:    

View more about this paper here: <https://typeset.io/papers/single-reference-frame-multiple-current-macroblocks-scheme-9wtmi5defu>

Single Reference Frame Multiple Current Macroblocks Scheme for Multiple Reference Frame Motion Estimation in H.264/AVC

Tung-Chien Chen, Chuan-Yung Tsai, Yu-Wen Huang, and Liang-Gee Chen, *Fellow, IEEE*

Abstract—Due to the multiple reference frame motion estimation (MRF-ME), an H.264/AVC encoder requires ultrahigh memory bandwidth. Conventional multiple reference frames single current macroblock (MRSC) scheme only considers the data reuse within one frame, and requires on-chip memory size and off-chip memory bandwidth in proportional to the reference frame number. In this paper, a single reference frame multiple current macroblocks (SRMC) scheme is presented to further exploit the data reuse at frame level. With frame-level rescheduling of the motion estimation ME procedures in different reference frames, one loaded search window can be utilized by multiple current MBs in different original frames. The demanded on-chip memory size and off-chip memory bandwidth for MRF-ME can thus be reduced to those supporting only one reference frame. Moreover, based on SRMC scheme, an architecture prototype with two-stage mode decision flow is proposed. For HDTV specifications, 62.21 KB (74.8%) of SRAM and 364.3 MB/s (62.6%) of system bandwidth are saved in comparison with the MRSC scheme.

Index Terms—ISO/IEC 14496-10 AVC, ITU-T Rec. H.264, JVT, motion estimation, multiple reference frame, VLSI architecture.

I. INTRODUCTION

THE H.264/AVC video coding standard [1], [2] can save 25%–45% and 50%–70% of bitrates when compared with MPEG-4 advanced simple profile and MPEG-2, respectively [3]. The coding gain mainly comes from the new prediction tools. The enormous computation and ultrahigh memory bandwidth are the penalties. The instruction profiling shows that 2.76 tera-operations per second (TOPS) of computational load and 4.25 tera-bytes/s (TB/s) of memory access are required to real-time encode SDTV (YUV420, 720×480 , 30 fps) videos (JM8.5 [4], baseline options, full search, four reference frames, and search range of $[-32, +31]$). The inter prediction occupies over 95% of the computational resource, which is mainly caused by multiple reference frames motion estimation (MRF-ME) [5].

For MRF-ME, the high computation complexity and large system bandwidth requirement are two main challenges for hardware implementation. Many fast algorithms have been proposed to decrease the computational complexity with a negligible loss of video quality for inter mode decision [6], [7] and for MRF-ME [8], [9]. However, these techniques are more suited for software implementation than hardware implementation. As for VLSI hardware architecture, many researchers take

more focuses on designing parallel architectures for the high throughput or low power issues [10]–[13]. After the parallel processing, the problem of high computation complexity can be solved. For a VLSI hardware system where the system bandwidth is limited, the following challenge is to reduce the large requirement of system bandwidth for MRF-ME.

In tradition, the data of current macroblock (MB) and search window (SW) are loaded from system memory and then buffered in on-chip SRAMs or registers. By means of local data reuse (DR), the system bandwidth requirement can be greatly reduced. Four DR strategies [14], [15] were proposed with different tradeoffs between system bandwidth and local memory size for previous standards. However, these schemes cannot efficiently support MRF-ME in H.264/AVC. The required system bandwidth and local memory size are linearly increased with the number of reference frames.

In this paper, we propose a new frame-level DR scheme. With the frame-level rescheduling, the data of one loaded SW can be reused by multiple current MBs in different original frames for MRF-ME, and the system bandwidth is greatly reduced. In addition, the required local memory size is almost the same with the previous design supporting only one reference frame.

The rest of this paper is organized as follows. A review of the H.264/AVC inter prediction and the conventional DR scheme is described in Section II. In Section III, the concepts of frame-level DR and rescheduling are described. In Section IV, we proposed the framework of SRMC including the two-stage mode decision flow and the corresponding architecture. The quality evaluation as well as performance evaluation are presented in Section V. Finally, Section VI gives a conclusion.

II. FUNDAMENTALS AND PROBLEM STATEMENT

A. Inter Prediction in H.264/AVC

In H.264/AVC, Inter prediction supports variable block size ME (VBS-ME) and MRF-ME. The MRF-ME allows using more than one reconstructed frame in both forward and backward directions for the temporal prediction. It is very efficient for uncovered backgrounds, repetitive motions, highly-textured areas, and etc. [8]. For VBS-ME, each MB can be split up in four kinds of partitions: 16×16 , 16×8 , 8×16 , and 8×8 . If the partition 8×8 is selected, each block can be further split into four kinds of subpartitions: 8×8 , 8×4 , 4×8 , and 4×4 . This tree-structured partition has 41 different blocks within one MB and gives rise to a large number of possible combinations.

The mode decision algorithm is not specified in the H.264/AVC standard and is left as an open issue. In the reference software [4], two kinds of Lagrangian mode decisions are supported—the high-complexity and the low-complexity mode decision. Because of the sequential issue, the high-complexity mode decision is less suitable for the hardware implementation.

Manuscript received December 9, 2005; revised July 13, 2006. This work was supported in part by the National Science Council, Taiwan, R.O.C., under Grant 95PFA0106257. This paper was recommended by Associate Editor M. Comer.

The authors are with the DSP/IC Design Laboratory, Department of Electrical Engineering and Graduate Institute of Electronics Engineering, National Taiwan University, Taipei 10617, Taiwan, R.O.C. (e-mail: djchen@video.ee.ntu.edu.tw; cytsai@video.ee.ntu.edu.tw; yuwen@video.ee.ntu.edu.tw; lgchen@video.ee.ntu.edu.tw).

Digital Object Identifier 10.1109/TCSVT.2006.887130

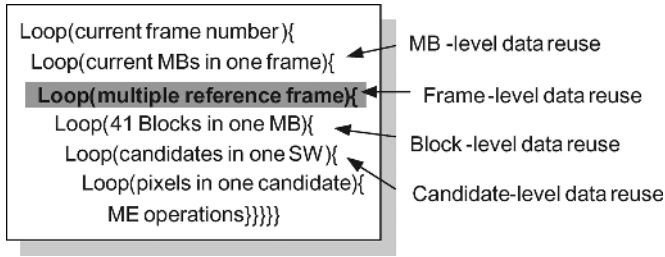


Fig. 1. Operation loops of MRF-ME for H.264/AVC.

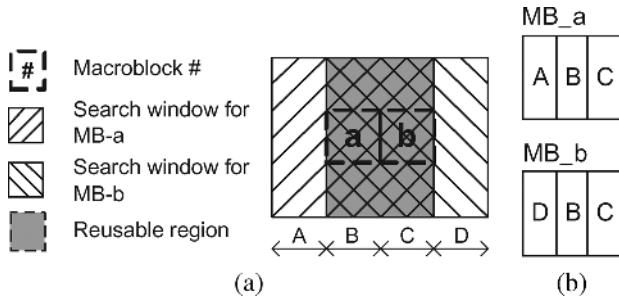


Fig. 2. Level-C data reuse scheme. (a) There are overlapped region of SWs for horizontally adjacent MBs. (b) Physical location to store SW data in local memory.

As for the low-complexity mode decision, the Lagrangian cost function considers both the distortion and the rate parts. The distortion is evaluated by sum of absolute differences (SAD) or sum of absolute transformed differences (SATD) between predictors and original pixels. The rate is estimated by the number of bits required to code the reference frame number and the motion vectors (MVs).

B. Conventional Data Reuse Scheme and Problem Statement

In ME, in order to find the best matched candidate of one current MB, a SW for each reference frame has to be searched. The bandwidth between system memory and ME core is very heavy if all required pixels are loaded from system memory. It consumes too much power and is not achievable in the platform-based system where the system bandwidth is limited. A common solution is to design local buffers to store reusable data. By means of local memory access, the system bandwidth can be greatly reduced.

Fig. 1 shows the operation loops with VBS-ME and MRF-ME for H.264/AVC. In the second loop, the SWs of the neighboring current MBs for one reference frame are considerably overlapped. So are the reference pixels of neighboring candidates for one current MB in the fifth loop. Four DR schemes, indexed from level-A to level-D [14], [15], have been proposed with different tradeoffs between local memory size and system bandwidth. Level-A requires the smallest local memory size and the highest system bandwidth, while level-D has the largest local memory size and the lowest system bandwidth. Fig. 2(a) describes the level-C DR scheme, which is generally adopted nowadays and will be used as the benchmark to explain our framework. The level-C scheme reuses the horizontally overlapped region between two SWs of the neighboring current MBs. As Fig. 2(b) shows, the SW of MB-b includes the area

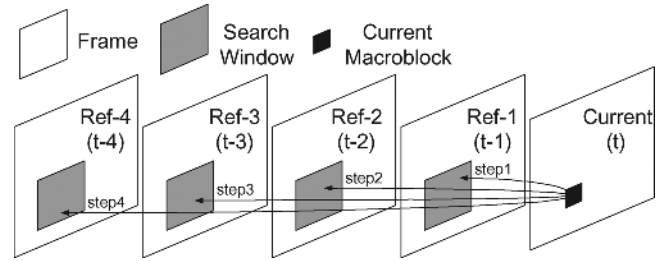


Fig. 3. MRSC scheme for MRF-ME requires multiple SWs memories. The reference pixels of multiple reference frames are load independently according to the level-C data reuse scheme.

TABLE I
SYSTEM BANDWIDTH AND SW MEMORY SIZE FOR MRF-ME
WITH LEVEL-C MRSC SCHEME

System Bandwidth (MByte/Sec)		Local Memory Size (KByte)	
MPEG-1/2/4 (Ref.=1)	H.264 (Ref.=4)	MPEG-1/2/4 (Ref.=1)	H.264 (Ref.=4)
60.13	209.43	6.656	25.86
SDTV (720x480, 30fps), Search range = [-32,+31]			

of B , C , and D . When the ME of MB-a is finished and the MB-b is going to be processed, only the reference pixels in D are loaded to replace A in the local memory.

In H.264/AVC, to support MRF-ME with level-C DR scheme, multiple SW memories are implemented as shown in Fig. 3. There are four SW memories, and each SW memory will be independently loaded and updated as Fig. 2. This can be referred as multiple reference frames single current MB (MRSC) scheme. Table I shows the requirement of system bandwidth and local memory size for MRSC scheme. The hardware cost is almost proportional to the maximum reference frame number. The large system bandwidth requirement increases the power consumption and becomes the bottleneck of the whole system. In addition, the increased memory size increases the silicon area and cost. Recently, the block-level data reuse scheme can be utilized for the fourth loop in Fig. 1 [10], [11]. The distortion costs of the smallest 4×4 blocks can be computed first. Then, the costs of larger blocks can be on-line calculated by summing up the corresponding 4×4 costs. However, this reuse scheme can only reduce the computation complexity but the system bandwidth. A new DR scheme is demanded for the MRF-ME loop.

III. PROPOSED DATA REUSE SCHEME

In this section, a single reference frame multiple current MBs (SRMC) scheme is proposed. To further reduce system bandwidth and local memory requirement, the SRMC scheme can exploit the frame-level DR in the multiple reference frame loop as shown in Fig. 1. Note that the proposed SRMC scheme is orthogonal to traditional candidate-level and MB-level DR schemes. That is, the SRMC scheme can co-work with any of the four conventional DR schemes.

A. Frame-Level Data Reuse

Fig. 4 shows the concept of frame-level DR in SRMC scheme. The reconstructed frame at time slot $t-4$, is the first reference frame of the original frame at time slot $t-3$. It is also the second,

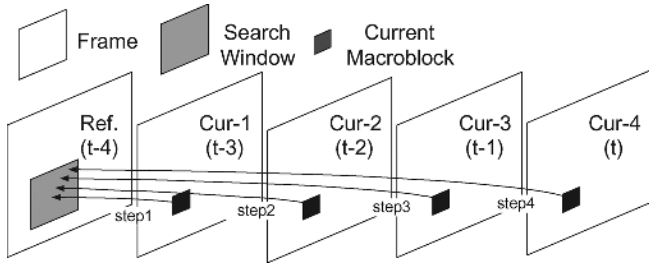


Fig. 4. SRMC scheme can exploit the frame-level DR for MRF-ME. Only single SW memory is required.

third, and fourth reference frame of original frames at time slot $t - 2$, $t - 1$, and t , respectively. Therefore, when the SW in the first previous reconstruction frame of one current MB is loaded to local memory, it can also be reused by the current MBs at the same location of the following original frames.

In MRSC scheme, one current MB is loaded only one time, and one reference SW is loaded several times. In SRMC scheme, one current MB is loaded several times while one reference SW is only loaded once. Since the SW is much larger than one MB, both the bandwidth and memory size can be largely reduced. When the frame size and the search range (SR) are increased, the benefit of the SRMC scheme will become more obvious. For HDTV video specifications, after the SRMC scheme is applied for MRF-ME, the required hardware resource is almost the same with the hardware resource supporting only one reference frame.

B. Frame-Level Rescheduling

In order to achieve frame-level DR, the ME procedures of each current MB in different reference frames are processed at different time slots. Therefore, the frame-level rescheduling is designed for the first three loops in Fig. 1. Fig. 5 shows the original schedule of the MRSC scheme and the rearranged schedule of the SRMC scheme for MRF-ME. It is assumed that there are six MBs in each frame and five P-frames to be coded. The maximum reference frame number is four. Three indexes are used to explain one ME cube—a ME procedure of one current MB for one reference frame. The first, second, and third indexes stand for the absolute time information of the processed current MB, the absolute time information of the corresponding SW, and the current MB index in one original frame, respectively. The ME cubes are processed sequentially along *Ref. frame plane*, *MB schedule*, and then *Frame schedule* axes.

Fig. 5(a) represents the original schedule of MRSC scheme. The ME cubes are performed reference frame by reference frame for one current MB, MB by MB in one frame, and then frame by frame, just like the reference software [4]. The MB cubes in one vertical column stand for the ME tasks of one current MB in different reference frames. According to the indices, the first, second, third and fourth ME cubes in one column represent the *step-1*, *step-2*, *step-3*, and *step-4* searching processes in Fig. 3. Note that the ME cubes in one column have the same t index and different f index.

Fig. 5(b) represent the rearranged schedule for SRMC scheme. The second, third, and fourth horizontal rows of each

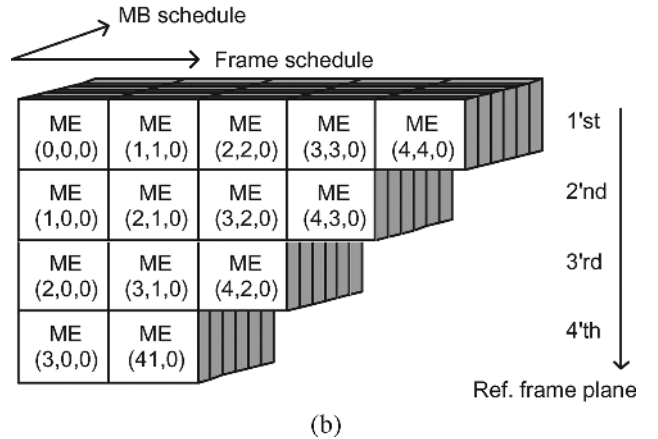
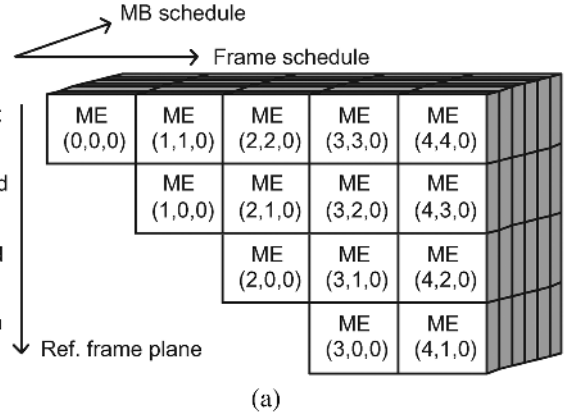
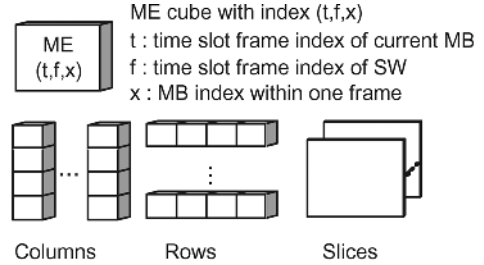


Fig. 5. Schedule of MB tasks for MRF-ME. (a) Original (MRSC) version. (b) Proposed (SRMC) version.

slice are shifted leftward for one, two, and three frame slots, respectively. Now, the first, second, third and fourth ME cubes in one vertical column represent the *step-1*, *step-2*, *step-3*, and *step-4* searching processes in Fig. 4. The ME cubes in one vertical column have the same index of f and different index of t . Therefore, one SW data can be reused by multiple current MBs for different reference frames, and the MRF-ME is still successively achieved.

IV. PROPOSED FRAMEWORK FOR SRMC SCHEME

A. Mode Decision for SRMC Scheme

The problem of inaccurate mode decision will occur after the block-level data reuse with the parallel hardware and frame-level rescheduling with the proposed SRMC scheme. In reference software, the Lagrangian cost function takes MV costs into consideration. The MV of each block is generally predicted by the medium value of MVs from the left, top, and

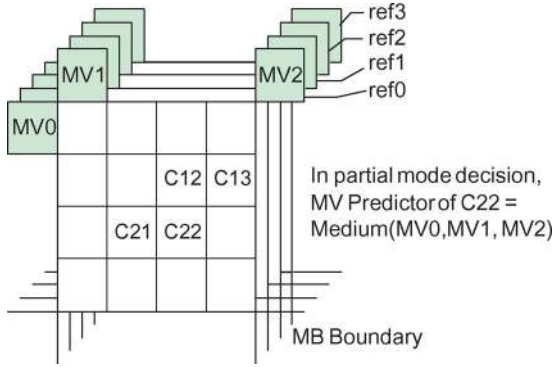


Fig. 6. Estimated MVPs in PMD for Lagrangian mode decision.

TABLE II
TWO-STAGE MODE DECISION

	Partial Mode Decision (PMD)	Final Mode Decision (FMD)
Process	On-line/Dedicated hardware	Off-line/System RISC
Task	decide the best matches of 41 blocks in each reference frame	Decide the best combination from all block modes in all reference frames

top right neighboring blocks. Not until the modes of neighboring blocks and MBs are decided can the motion vector predictor (MVP) of the current block or MB become available. Therefore, the rate term of the Lagrangian cost function can be computed only after that MVs of the neighboring blocks or MBs are decided from the solution space of variable blocks and multiple reference frames.

To support the block-level data reuse scheme in hardware, the distortion values of all candidates for all variable blocks must be stored for the following Lagrangian mode decision flow, which is completely impossible due to its extremely large data size. The modified MVP [16] is adopted for all 41 blocks in one MB as shown in Fig. 6. The exact MVPs of variable blocks, which are ideally the medium of MVs of the left, top, and top-right blocks, are changed to the medium of MVs of the left, top, and top-right MBs. For example, the exact MVP of the C22 4×4 -block is the medium of the MVs of C12, C13, and C21. We change the MVPs of all 41 blocks to the medium of MV0, MV1, and MV2 in order to facilitate the parallel processing with block-level data reuse.

Furthermore, in the SRMC scheme, the ME tasks in different reference frames are processed at different time slots. The MV information in different reference frames cannot be jointly considered for the extremely large storage requirement. We propose a two-stage mode decision method to deal with this problem. The mode decision flow is divided into partial mode decision (PMD) and final mode decision (FMD) as shown in Table II. The PMD on-line decides the best matches of 41 variable blocks for each reference frame. These MVPs are estimated according to the information within one reference frame as shown in Fig. 6. The MVs and the distortion costs of these suboptimal results are written to the external memory. After the PMD results of all reference frames are generated for a certain current MB, the FMD decides the best combination of variable blocks in different reference frames with system RISC. At FMD, the exact MV costs can be sequentially generated.

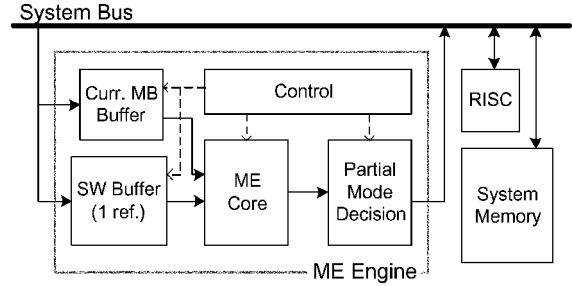


Fig. 7. Proposed architecture with SRMC scheme.

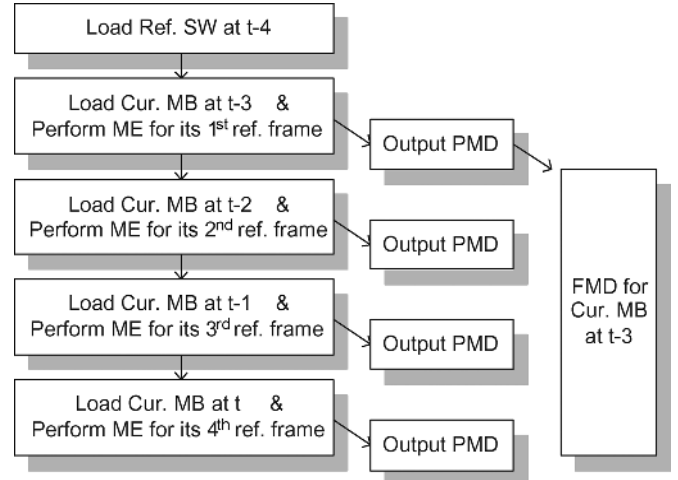


Fig. 8. Schedule of SRMC scheme in the proposed framework.

B. Architecture Design

Fig. 7 shows the architecture of H.264/AVC ME engine using the SRMC scheme. Different from the MRSC scheme, only one SW memory is required to support MRF-ME. The ME core computes the candidates' distortion values, and the PMD engine on-line decides the MVs of variable blocks according to the estimated MVPs. Both the full-search or fast-search ME algorithms can be implemented as the ME core. The PMD results are buffered at system memory, and then the RISC performs FMD.

Fig. 8 shows the schedule. Referred to Fig. 4, the SW at the frame $t-4$ is loaded to SW buffer first. Then, the ME task of the current MB in the frame $t-3$ will be performed. After that, the FMD of this current MB is then done by RISC after the PMD results are written out. At the same time, the current MBs at the same location of the following frames marked as $t-2$, $t-1$, and t are processed one after another. Although multiple current MBs are loaded, only one current MB buffer is required. Note that FMD can also be implemented as dedicated hardware with the same schedule. In this case, another system bandwidth to load back the PMD results is the penalty.

V. SIMULATION RESULTS AND PERFORMANCE EVALUATION

A. Simulation Results

Fig. 9 shows the rate-distortion efficiency of the reference software and the proposed framework. Four sequences with different characteristics are used for the experiment. Foreman has lots of deformations and media motions. Mobile has

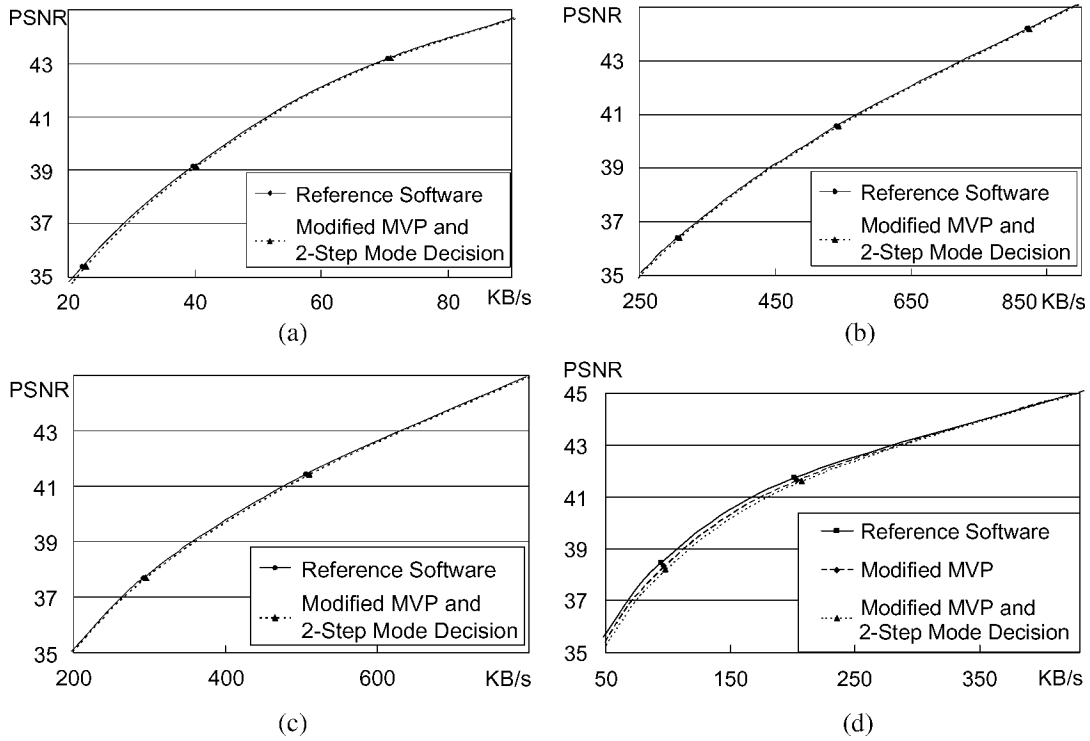


Fig. 9. Rate-distortion efficiency of the reference software and the proposed framework. Four sequences with different characteristics are used for the experiment. Foreman has lots of deformation with media motions. Mobile has complex textures and regular motion. Akiyo has the still scene, while Stefan has large motions. The encoding parameters are Baseline profile, IPPP... structure, CIF, 30 frames/s, 4 reference frames, ± 16 -pel search range, and low complexity mode decision.

TABLE III
PERFORMANCE OF THE PROPOSED SRMC SCHEME

	System Bandwidth (MByte/Sec)			Local Memory Size (KByte)		
	MRSC Scheme	SRMC Scheme	Saved (%)	MRSC Scheme	SRMC Scheme	Saved (%)
Case A	46.97	25.63	-35.4 %	11.78	2.56	-78.3 %
Case B	209.43	97.21	-53.6 %	25.86	6.65	-74.3 %
Case C	973.82	364.28	-62.6 %	83.20	20.99	-74.8 %

Case A : CIF (352x288, 30fps), Search range = [-16,+15], 5 reference frames

Case B : SDTV (720x480, 30fps), Search range = [-32,+31], 4 reference frames

Case C : HDTV (1280x720, 30fps), Search range = [-64,+63], 4 reference frames

complex textures and regular motions. Akiyo is a still scene, while Stefan has large motions. The encoding parameters are Baseline profile, IPPP... structure, four reference frames, ± 16 -pel search range, and low complexity mode decision. In the proposed framework, two algorithm modifications are involved compared with reference software. First, to facilitate the parallel processing for VBS-ME, the modified MVP is adopted for the block-level data reuse. Second, to reduce the system bandwidth for MRF-ME, the SRMC scheme are proposed with two-stage mode decision for the frame-level data reuse.

According to the simulation results, the proposed framework has similar compression performances compared to the reference software except for Foreman. For Foreman sequence in lower bitrates, up-to 0.3 dB quality drop is induced compared with the reference software. This is mainly caused by the difference between estimated MVP and exact MVP. However, compared to the previous hardwired encoder where the modified MVP is applied [16], only 0.12 dB quality drop is induced by the proposed two-stage mode decision. In addition, for applications with higher bitrates, the distortion part dominates the

Lagrangian cost function, and the compression performance of the proposed framework approaches the idea case.

B. Performance Evaluation

In this subsection, we will show the efficiency of the proposed SRMC DR scheme in terms of system bandwidth and memory size requirement. The level-C DR scheme is used as original MRSC DR scheme for the comparison. The framework with two-stage mode decision flow is used. The PMD is done by the dedicated hardware while the FMD is handled by the RISC. The required bus bandwidth and memory size requirement of the MRSC schedule are calculated as follows:

$$BW_{MRSC} = BW_{one_ref} \times num_ref + BW_{one_cur_MB}$$

$$mem_{MRSC} = mem_{one_SW} \times num_ref + mem_{one_cur_MB}$$

The required bus bandwidth and memory size of the rearranged SRMC schedule are calculated as follows:

$$BW_{SRMC} = BW_{one_ref} + BW_{one_cur_MB} \times num_ref + BW_{PMD}$$

$$mem_{SRMC} = mem_{one_SW} + mem_{one_cur_MB}.$$

The BW_{PMD} includes the MVs and the matching costs of variable blocks. The bus bandwidth requirement of BW_{PMD} is relatively small. Table III summarizes the performance evaluation for three cases. The proposed SRMC scheme can save about 75% of on-chip memory size and 35.4%–62.6% of system bandwidth compared to the conventional MRSC scheme. The BW_{one_ref} increases with larger search range. Therefore, the proposed SRMC scheme has better performance for the videos with larger frame sizes that inherently require larger search ranges.

VI. CONCLUSION

In this paper, we proposed a simple and effective technique to reduce the external system bandwidth and internal memory size for MRF-ME. By frame-level rescheduling, the procedures for multiple current MBs in different original frames can simultaneously utilize the data of single SW. In the proposed framework, the SRMC scheme reduces not only 63% of external system bandwidth but also 75% of internal memory size for HDTV specifications.

REFERENCES

- [1] *Draft ITU-T Rec. Final Draft Int. Standard of Joint Video Specification*, ITU-T Rec. H.264 and ISO/IEC 14496-10 AVC, May 2003.
- [2] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003.
- [3] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G. J. Sullivan, "Rate-constrained coder control and comparison of video coding standards," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 688–703, Jul. 2003.
- [4] Joint Video Team Reference Software JM8.5 Sep. 2004 [Online]. Available: <http://bs.hhi.de/suehring/tml/download/>
- [5] T. Wiegand, X. Zhang, and B. Girod, "Long-term memory motion-compensated prediction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 1, pp. 70–84, Feb. 1999.
- [6] P. Yin, H.-Y. Tourapis, A. Tourapis, and J. Boyce, "Fast mode decision and motion estimation for JVT/H.264," in *Proc. Int. Conf. Image Processing*, 2003, pp. III-853–III-856.
- [7] P. Zhang, Y. He, S. Yang, and Y. Zhong, "Performance and complexity joint optimization for H.264 video coding," in *Proc. Int. Symp. Circuits Syst.*, 2003, pp. II-888–II-891.
- [8] Y. Su and M.-T. Sun, "Fast multiple reference frame motion estimation for H.264/AVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 3, pp. 447–452, Mar. 2006.
- [9] Y.-W. Huang, B.-Y. Hsieh, T.-C. Wang, S.-Y. Chien, S.-Y. Ma, C.-F. Shen, and L.-G. Chen, "Analysis and reduction of reference frames for motion estimation in MPEG-4 AVC/JVT/H.264," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2003, pp. 145–148.
- [10] Y.-W. Huang, T.-C. Wang, B.-Y. Hsieh, and L.-G. Chen, "Hardware architecture design for variable block size motion estimation in MPEG-4 AVC/JVT/ITU-T H.264," in *Proc. IEEE Int. Symp. Circuits Syst.*, 2003, pp. 796–799.
- [11] S. Y. Yap and J. V. McCanny, "A VLSI architecture for variable block size video motion estimation," *IEEE Trans. Circuit Syst. II*, vol. 51, no. 7, pp. 384–389, Jul. 2004.
- [12] J.-H. Lee and N.-S. Lee, "Variable block size motion estimation algorithm and its hardware architecture for H.264," in *Proc. IEEE Int. Symp. Circuits and Systems*, May 2004, vol. 3, pp. 740–743.
- [13] T.-C. Chen, Y.-H. Chen, S.-F. Tsai, and L.-G. Chen, "Architecture design of low power integer motion estimation for H.264/AVC," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Process.*, 2006, pp. III-900–III-903.
- [14] M. Y. Hsu, "Scalable Module-Based Architecture for MPEG-4 BMA Motion Estimation," M.S. thesis, Graduate Inst. Electron. Eng., National Taiwan Univ., Taipei, Taiwan, R.O.C., 2000.
- [15] J.-C. Tuan, T.-S. Chang, and C.-W. Jen, "On the data reuse and memory bandwidth analysis for full-search block-matching VLSI architecture," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 1, pp. 61–72, Jan. 2002.
- [16] T.-C. Chen, S.-Y. Chien, Y.-W. Huang, C.-H. Tsai, C.-Y. Chen, T.-W. Chen, and L.-G. Chen, "Analysis and architecture design of an HDTV720p 30 frames/s H.264/AVC encoder," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 6, pp. 673–688, Jun. 2006.