

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

Tung-Chien Chen, Yu-Wen Huang, Chuan-Yung Tsai, Chao-Tsung Huang, and Liang-Gee Chen
DSP/IC Design Lab, Graduate Institute of Electronics Engineering and Department of Electrical Engineering
National Taiwan University, Taipei, Taiwan; Email: djchen@video.ee.ntu.edu.tw

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

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. The coding gain mainly comes from new prediction tools. However, enormous computation and ultra high memory bandwidth are the penalties. The instruction profiling shows that 2.76 tera-operations/s (TOPS) of computational loading and 4.25 tera-bytes/s (TB/s) of memory access are required for real-time encoding SDTV (YUV420, 720x480, 30fps) videos (JM8.5 [3], baseline options, full search, four reference frames, search range [-32, +31]). Among all encoding processes, inter prediction occupies 99% of computation and memory access. This is mainly resulted from multiple reference frames motion estimation (MRF-ME) [4].

The MRF-ME allows to use as many as five reference reconstructed frames in both temporal directions. It is very effective for uncovered backgrounds, repetitive motions, highly textured areas, etc [5]. Many fast algorithms [5][6][7][8] have been proposed to decrease the computational complexity without significant loss of video quality. However, for platform-based VLSI designs in which the high computation requirement can be easily solved by increasing the parallelism of processing elements, the real challenge is to reduce the SRAM size and bus bandwidth. Usually, current macroblock (MB) data and search window (SW) data are buffered in on-chip SRAMs or registers to reduce the external memory access. Four data reuse strategies [9][10] were proposed with different tradeoffs

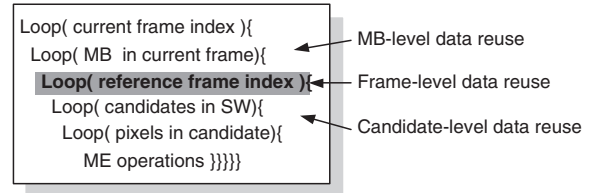


Fig. 1. The operation loops of MRF-ME for H.264/AVC.

between bus bandwidth and local memory size. Even so, the previous scheme cannot efficiently support the MRF-ME. The required bus bandwidth and local memory size are linearly increased with the number of reference frames. In this paper, we propose a new data reuse scheme to achieve MRF-ME with almost the same local memory requirements as the previous scheme supporting only one reference frame. With frame-level rescheduling, one SW data can be reused by multiple current MBs in different frames.

The rest of this paper is organized as follows. In Section II, the conventional data reuse scheme is reviewed. In Section III, the concepts of frame-level data reuse and MB rescheduling are described. The H.264/AVC encoding framework is proposed in Section IV, and evaluation results as well as discussions are presented in Section V. Finally, Section VI gives a conclusion.

II. CONVENTIONAL DATA REUSE SCHEME

In motion estimation (ME), in order to find the best matched candidate and its corresponding motion vector (MV), a SW within one reference frame has to be searched. The traffic between frame buffer and ME core is very heavy (in the order of TB/s for SDTV videos). It consumes too much power and is not achievable in today's VLSI technology. The common solution is to design local buffers to store reusable data. By means of local memory access, the external memory bandwidth can be greatly reduced. Figure 1 shows the operation loops of MRF-ME for H.264/AVC. In the fourth loop, pixels in neighboring candidate blocks of current MB are considerably overlapped, and so are the SWs of neighboring current MBs in the second loop. Four data reuse strategies have been proposed with different tradeoffs between local memory size and system bus bandwidth, and are indexed from level-A to D [9][10]. Level-A requires the smallest local memory size and

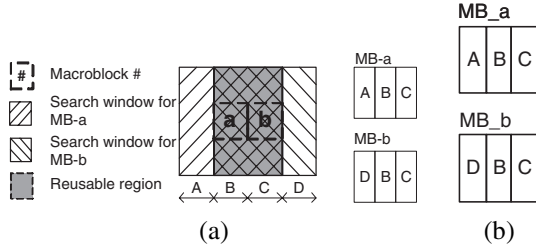


Fig. 2. Level-C data reuse strategy; (a) overlapped region of SWs for horizontally adjacent MBs. (b) the physical location of loaded SW data in local memory.

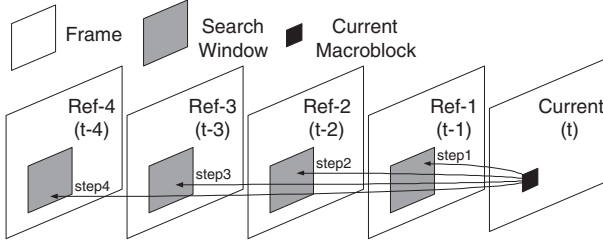


Fig. 3. Illustration of MRSC scheme requiring multiple SWs memories to achieve MRF-ME.

the highest external bandwidth, while level-D has the largest local memory size and the lowest external bandwidth. Figure 2(a) describes the level-C data reuse strategy, which is usually adopted in nowadays and will be used as examples to explain our framework. For storing one SW and one current MB data, some local buffers are required. Since the neighboring SWs of MB-a and MB-b have a large overlapped area (B and C in Fig. 2(a)), when MB-b is processed, only the data of area D are loaded to replace those of area A in the local memory, as shown in Fig. 2(b).

In H.264/AVC, MRF-ME allows to use more than one reference frames, as shown in Fig. 3. To support MRF-ME with level-C data reuse strategy, multiple SW memories for MRF can be implemented, and each SW memory will be loaded as expressed in Fig. 2. This can be referred as a multiple reference frames single current macroblock (MRSC) scheme. The requirement of bus bandwidth and memory size with level-C MRSC scheme is shown in Table I. The hardware cost is nearly proportional to the maximum reference frame number. In our experience, for a full-search ME accelerator, the area of SW memories is similar to that of logic gates. When fast block matching is adopted, the SW memories will dominate the entire silicon area, especially for MRF-ME. Hence, a new data reuse scheme is urgently demanded.

III. PROPOSED DATA REUSE SCHEME

In this section, a single reference frame multiple current macroblocks (SRMC) data reuse scheme is proposed. SRMC further exploits the data reuse at frame-level. Please note that SRMC is orthogonal to traditional candidate-level and MB-level data reuse strategies. That is, our scheme can be integrated with any of the four conventional strategies. By rescheduling of the operation loops in Fig. 1, SRMC can be

TABLE I
REQUIRED BUS BANDWIDTH AND SRAM SIZE FOR MRF-ME WITH LEVEL-C MRSC SCHEME.

Bandwidth (Mbytes/Sec)		Local Buffer (Kbytes)	
MPEG1/2/4 (Ref = 1)	H.264 (Ref = 4)	MPEG1/2/4 (Ref = 1)	H.264 (Ref = 4)
60.13	209.43	6.656	32.26

SDTV (720x480 30fps), SR = [-32,+31]

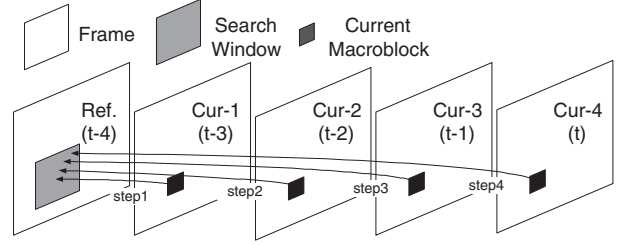


Fig. 4. Illustration of SRMC scheme requiring only single SW memory to achieve MRF-ME.

applied and the MRF-ME can be achieved with significantly reduced bus bandwidth and memory size.

A. Frame-Level Data Reuse

Figure 4 shows the concept of frame-level data reuse in SRMC. 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, 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 frame of the current MB is loaded to local memory, it can also be utilized by the MBs at the same location of the following frames. In other words, the ME procedures of one current MB for different reference frames are spread and processed at different time slots. In this way, one current MB is loaded several times while one reference SW only needs to be loaded once. Since the SW is much larger than one MB, both the bus bandwidth and memory size are about the same as those of MRSC scheme supporting only one reference frame.

B. Frame-Level Rescheduling

Figure 5 shows the original schedule and rearranged schedule of the first three loops in Fig. 1. for MRF-ME. It is assumed that there are four MBs in each frame, four reference frames for each MB, and four P-frames to be coded. In Fig. 5(a), three indices are used to explain the ME procedures. The first, second, and third indices stand for the absolute time information of current MB, the absolute time information of the corresponding SW (reference frame), and the location of current MB, respectively. A vertical column of multiple slices (with different depths) denotes one frame task for all MBs of the same frame. The block matching process is performed reference frame by reference frame, MB by MB, and then frame by frame. As shown in Fig. 5(b), the rearranged

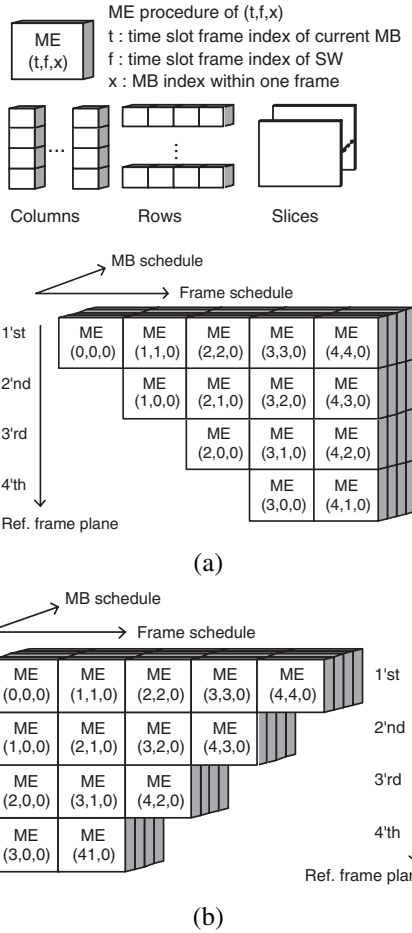


Fig. 5. Schedule of MB tasks for MRF-ME; (a) the original (MRSC) version; (b) the proposed (SRMC) version.

TABLE II
DESCRIPTION OF THE TWO-STAGE MODE DECISION.

	Partial mode decision	Final mode decision
Process	On-line/Dedicated hardware	Off-line/System RISC
MV Cost	Estimated	Precise
Task	Separately decide the best matches of 41 blocks in each reference frame.	Decide the best combination from all block modes in all reference frames.

schedule shifts the second, third, and fourth horizontal rows of multiple slices leftward (at frame-level) for one, two, and three frame slots, respectively. In this way, multiple current MBs of different frames can share the same SW data, and the MRF-ME is still successively achieved.

IV. PROPOSED SYSTEM ARCHITECTURE WITH SRMC

In H.264/AVC reference software, the Lagrangian mode decision [2] is adopted. The Lagrangian mode decision takes MV costs into account, which improves the coding performance significantly but causes data dependencies between neighboring MBs and sub-blocks. As shown in Fig. 6, not

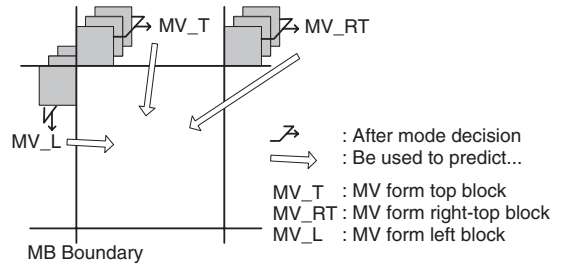


Fig. 6. Example of MVs of neighboring blocks required by Lagrangian mode decision.

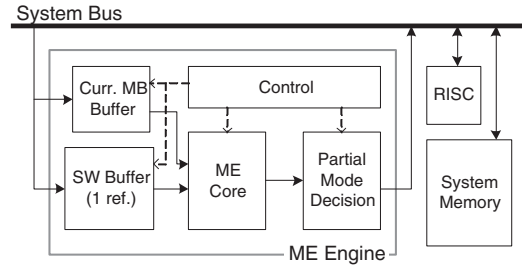


Fig. 7. Proposed system architecture with SRMC scheme.

until the modes of neighboring MBs are decided can the motion vector predictor (MVP) of the current MB become available. This data dependency conflicts with the SRMC scheme. In the original MRSC scheme, the ME procedures for different reference frames of one current MB are at the same location of frame schedule axis, and the MB mode decision can be done without problems. Because the MBs of one frame are processed in raster order, the MV costs can be on-line calculated. However, in the proposed SRMC scheme, the ME procedures for different reference frames of one current MB are spread into different locations of frame schedule axis. The exact MVP of current MB can be calculated only when the block matching is done for the previous one reference frame. To add the MV costs with distortion for farther reference frames, all distortion values of candidates in farther reference frames must be stored, which is completely impossible.

We proposed 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 is responsible for separately on-line deciding the best matches of 41 blocks of an MB for each reference frame. Since the distortion costs of current MBs are available only for the previous one reference frame, the MVPs should be modified according to the limited available information. The MVs and the distortion costs of the sub-optimal matched candidates are moved to the external memory. After the PMD results for all reference frames of one current MB are generated, the FMD uses system RISC to decide the best configuration of block modes. At this time, the exact MV costs are used.

Figure 7 shows the system architecture of H.264/AVC ME engine using SRMC scheme. Different from MRSC scheme, only one SW memory is required to support MRF-ME. The

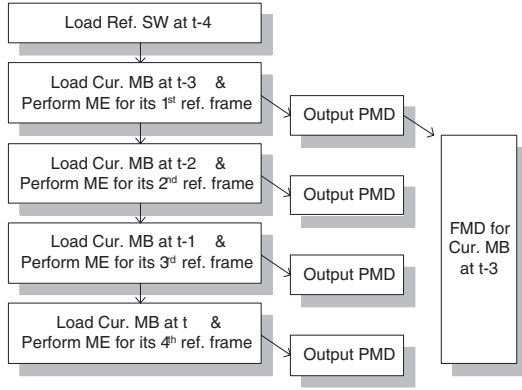


Fig. 8. The basic flow of SRMC in the proposed framework.

ME core computes the candidates' distortion values, and the PMD engine on-line decides the best MV of each sub-block. Full-search or fast ME algorithms can be implemented in the ME core. As stated before, the PMD results are buffered at external memory, and then the RISC performs FMD. Figure 8 shows the basic flow. Referred to Fig. 4, the SW at the frame marked as $t - 4$ is loaded first. Then, the ME procedure of the current MB in the frame marked as $t - 3$ will utilize the loaded SW data. The FMD of this current MB is then done by RISC after the PMD results are generated. 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. Therefore, although multiple current MBs are loaded, only one current MB buffer is required. Please note that FMD can also be implemented as dedicated hardware with the same schedule. The bus traffic of PMD results is an overhead for SRMC scheme.

V. PERFORMANCE EVALUATION AND DISCUSSION

In this section, the level-C data reuse strategy is used to evaluate the memory requirements without loss of generality. The mode decision of the original MRSC schedule and PMD of the rearranged SRMC schedule are done by dedicated hardware while the FMD of the rearranged SRMC schedule is handled by RISC. The required bus bandwidth and memory size of the original MRSC schedule are 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 follows.

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

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

Table III summarizes the evaluation for three cases. SRMC can save about 75% of on-chip buffer and 35.4%-62.6% of bus bandwidth. The bus bandwidth requirement of BW_{PMD} is quite small and independent of search range, while that of BW_{one_ref} increases with search range.

TABLE III
PERFORMANCE OF THE PROPOSED SRMC SCHEME.

	Bandwidth (Mbytes/Sec)			Local Buffer (Kbytes)		
	Original	Proposed	saved%	Original	Proposed	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

There still exist other issues for the SRMC scheme. First, some video quality will be lost due to the inaccurate MV costs in mode decision. More experiments are needed to find modified MV predictors with negligible quality loss. Second, SRMC scheme will enlarge the encoding latency in baseline profile from the order of MBs to the order of frames. Hence, it is more suitable for main profile applications, in which B-pictures are allowed with long encoding latency.

VI. CONCLUSION

In this paper, we proposed a simple and effective technique to reduce the external bus bandwidth and internal memory size for multi-frame motion estimation. By frame-level rescheduling, the procedures for multiple current macroblocks of different frames can simultaneously utilize the data of one single search window. The proposed system architecture reduces not only 63% of external bus bandwidth but also 75% of internal memory size for HDTV specifications.

REFERENCES

- [1] J. V. Team, *Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification*. ITU-T Rec. H.264 and ISO/IEC 14496-10 AVC, May 2003.
- [2] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G. J. Sullivan, "Rate-constrained coder control and comparison of video coding standards," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 688-703, July 2003.
- [3] *Joint Video Team Reference Software JM8.5*. <http://bs.hhi.de/~suehring/tml/download/>, 2004.
- [4] T. Wiegand, X. Zhang, and B. Girod, "Long-term memory motion-compensated prediction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, pp. 70-84, Feb. 1999.
- [5] Y. Su and M.-T. Sun, "Fast multiple reference frame motion estimation for h.264," in *Proc. of ICME*, 2004.
- [6] S. Wuytack, J. P. Diguët, F. V. M. Catthoor, and H. J. D. Man, "Formalized methodology for data reuse exploration for low-power hierarchical memory mappings," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 6, pp. 529-537, 1998.
- [7] E. D. Greef, F. Catthoor, and H. D. Man, "Program transformation strategies for memory size and power reduction of pseudoregular multimedia subsystems," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8, pp. 719-733, 1998.
- [8] Y. W. Huang, B. Y. Hsieh, T. C. W. ans 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. of ICASSP*, 2003.
- [9] M. Y. Hsu, *Scalable module-based architecture for MPEG-4 BMA motion estimation*. M.S. thesis, National Taiwan Univ., 2000.
- [10] 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 Transactions on Circuits and Systems for Video Technology*, vol. 12, pp. 61-72, Jan. 2002.