A Sub 100 mW H.264/AVC MP@L4.1 Integer-Pel Motion Estimation Processor VLSI for MBAFF Encoding

Yuichiro Murachi, Kosuke Mizuno, Junichi Miyakoshi, Masaki Hamamoto, Takahiro Iinuma,
Tomokazu Ishihara, Fang Yin, Jangchung Lee, Tetsuya Kamino, Hiroshi Kawaguchi, and Masahiko Yoshimoto
Department of Computer Science and Systems Engineering
Kobe University
Kobe, Japan

Abstract— **This paper describes a sub 100-mW H.264/AVC MP@L4.1 integer-pel motion estimation processor core for a low power video encoder. It supports macro block adaptive frame field (MBAFF) encoding and bi-directional prediction for a resolution of 1920 × 1080 pixels at 30 fps which haven't been realized by conventional methods[1][2]. The proposed processor core features a novel hierarchical algorithm, a reconfigurable ring-connected systolic array architecture, and a segmentation-free rectangle-access search window buffer. The processor core has been designed in a 90 nm CMOS technology, and its core size is 2.5 × 2.5 mm^2. With one core, one reference frame can be handled, and 48 mW is consumed at 1 V. Two-core configuration dissipates 96 mW for two reference frames.**

I. INTRODUCTION

In H.264/AVC, more than double workload of the conventional MPEG-2 is necessary for higher picture quality and lower bitrate [3]. Several H.264/AVC motion estimation (ME) processor cores have been developed to save the workload and power. ME is composed of an integer-pel ME (IME) and a fractional pel ME (FME). The IME finds integer-pel accuracy motion vectors (MVs), and then the FME calculates quarter-pel accuracy MVs using them. Because the workload of the IME accounts for over 80~90% of the ME, the power reduction in the IME is essential to reduce the total power of an ME processor. We propose the following three techniques to reduce the power in the IME keeping high picture quality:

- An IME algorithm that adopts a hierarchical and adaptive search method with image analysis, supporting MBAFF encoding.
- A reconfigurable ring-connected systolic array architecture that minimizes the amount of transferred pixel data, and reduces computational cycles.
- A search window buffer SRAM that enables instantaneous rectangular access, segmentation-free access [4] and sub-sampling access with smaller silicon area at lower power.

In this paper, the proposed algorithm, architecture and SWRAM is addressed in Section II, III, and IV, respectively. Then, these are followed by VLSI implementation and power estimation in Section V.

II. ALGORITHM

A. Algorithm Overview and Image Analysis

The proposed algorithm is a hierarchical search algorithm consisting of a coarse search and two fine searches. The flow chart is shown in Fig. 1.

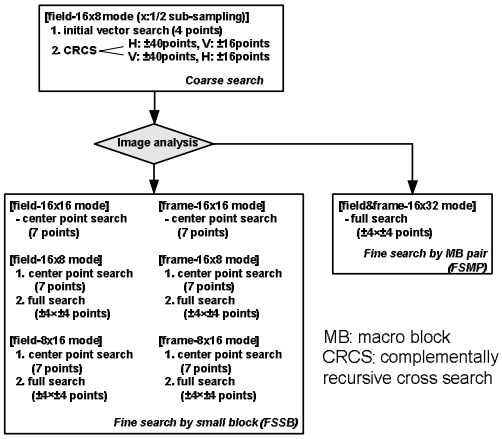

Fig. 1 IME algorithm flow.

The coarse search finds suboptimal MVs over a wide area in a search window, and analyzes the distribution of MVs for variable size of blocks. As the coarse search, we propose a new one-dimensional search method with low complexity and high picture quality, named complementally recursive cross search (CRCS). The method of the CRCS is illustrated in Fig. 2. In CRCS, 2 RCSs (recursive cross search) are executed in parallel with the same initial point. When search is over, the vector with smaller SAD between the two vectors of RCSs is chosen as result of CRCS.

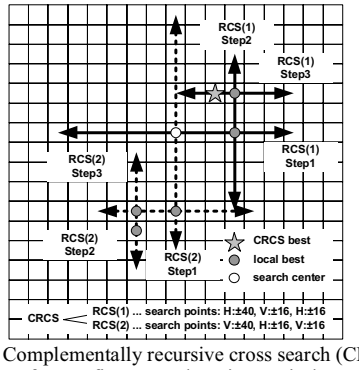

Fig. 2. Complementally recursive cross search (CRCS).

Then, either of two fine searches is carried out. The two fine searches comprise a fine search by small block (FSSB) and a fine search by macro block pair (FSMP), one of which is chosen according to the result of image analysis mentioned below. In FSSB, center point searches are carried out for 2 MBAFF modes and 3 block sizes. The purpose of the center point searches is to adjust the coarse

This work was supported by VLSI Design and Education Center (VDEC), the University of Tokyo in collaboration with Cadence Design Systems, Mentor Graphics and Synopsys, Inc.

search results to each mode's optimum. Then full searches are conducted to increase picture quality. In FSMP, a full search is conducted at MB pair size.

By using integer-pel MVs obtained from the coarse search in the field 16 × 8 block (Fig. 3), distribution of MVs is analyzed to reduce workload. The subscripts of the MVs give the block position (as upper or lower), the field of the template block (as top or bottom), and the field of the search window (as top or bottom). If both the temporal conditions ((1), (2)) and the spatial conditions ((3)–(6)) are under a threshold value (= 4), analysis result implies that the MVs are locally-distributed, and then the proposed search algorithm is branched to the FSMP. If any of (1)–(6) is not satisfied, the analysis result means that MVs are globally-distributed, and then the FSSB is conducted. By the above flow, the workload is reduced by 14%.

Temporal condition :

$$|MV_{Upper\_TT} - MV_{Upper\_BB}| < THR_{PATH} \quad (1)$$

$$|MV_{Lower\_TT} - MV_{Lower\_BB}| < THR_{PATH} \quad (2)$$

Spatial condition :

$$|MV_{Upper\_TT} - MV_{Lower\_BB}| < THR_{PATH} \quad (3)$$

$$|MV_{Upper\_TB} - MV_{Lower\_TB}| < THR_{PATH} \quad (4)$$

$$|MV_{Upper\_BT} - MV_{Lower\_BT}| < THR_{PATH} \quad (5)$$

$$|MV_{Upper\_BB} - MV_{Lower\_BB}| < THR_{PATH} \quad (6)$$

("| |" signifies a summation of H elements and V elements.)

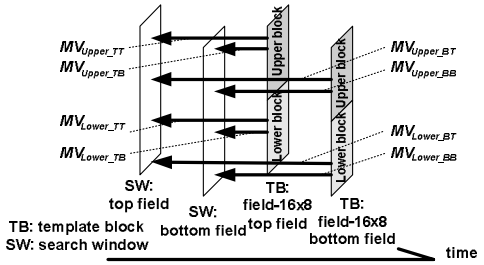

Fig. 3. Image analysis using MVs obtained from the coarse search.

B. Simulation Result

The parameters in the algorithm simulation are as follows: a resolution is HD (1920 × 1080 interlace), a frame rate is 30 fps, the number of reference frames are two for a P frame and two for a B frame, MBAFF is supported, and search range is ±128 × ±64. We used 13 test sequences including quick motion and high activity picture (Bronze with Credits, Church, European Market, Whale Show, Soccer Action, Track with Credits, Buildings along the Canal, View from Sky with Credits, Intersection, Streetcar, Yachting, Japanese Room, and Yacht Harbor). Fig. 4 shows comparisons of average workload and average PSNR over the above sequences between the proposed and conventional algorithms. The workload of the proposed algorithm is reduced to 5.2% and the picture degradation is only 0.047dB, compared with the conventional hybrid unsymmetrical cross multi hexagon grid search (UMHS) [5].

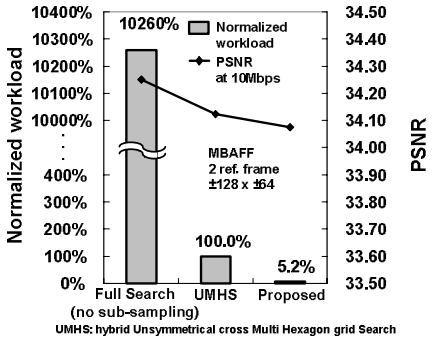

Fig. 4. Workload and PSNR of the proposed IME algorithm.

III. Architecture

The proposed IME algorithm employs three kinds of search methods (the full search (FS) and the CRCS described in the previous, plus one-time block matching on random access). In this section, a reconfigurable ring-connected systolic array (RRSA) is proposed, which supports the above three search methods efficiently at a less rate of data transfer and in fewer cycles.

A. Architecture Overview

Fig. 5 shows a block diagram of the proposed IME processor and the detail drawing of the RRSA inside. A search window buffer RAM (SWRAM: dual port = one read / one write) provides reference image data to the RRSA through a cross path circuit. The cross path circuit rotates pixel data on demand. A template-block buffer (TB buffer) is a register file that stores current image data.

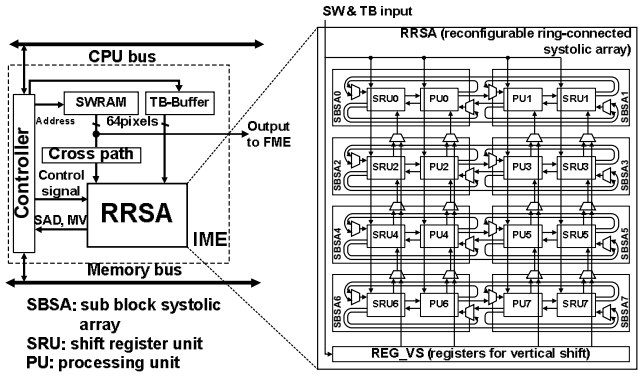

Fig. 5. A block diagram of the proposed IME processor core.

B. Reconfigurable Ring-Connected Systolic Array

The RRSA is comprised of eight sub block systolic arrays (SBSAs) and registers for vertical shift (REG_VS). Fig. 6 shows a block diagram of the SBSA. The SBSA consists of a PU and SRU. A PU consists of 8 × 8 processing elements (PEs), and a SRU consists of 8 × 8 shift register elements (SREs) as illustrated. A PE outputs absolute difference between an SW pixel and a TB pixel, while an SRE buffers a pixel for PE.

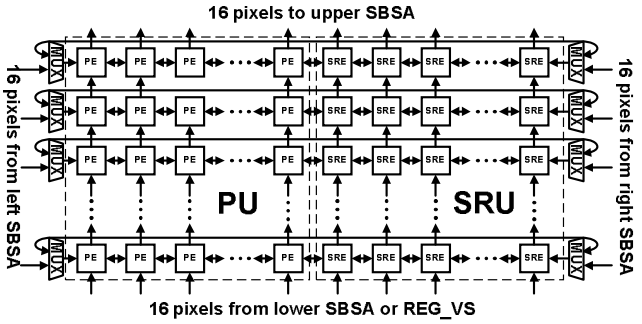

Fig. 6. A block diagram of the SBSA.

By changing a combination of SBSAs, PE arrays from 8 × 8 to 16 × 32 can be configured. Hence, SAD calculations in any block types (frame / field MBAFF modes, and 8 × 8 / 16 × 8 / 8 × 16 / 16 × 16 / 16 × 32 block sizes) can be performed by the proposed RRSA architecture.

The PU and SRU have a direct access path for initial load and buffered access path through REG_VS for FS. Furthermore the PU and SRU can continue to search at continuous points with less reload of pixels from the SWRAM by pixel reuse. The SBSA has an inter- and intra-connection (ring connection) with MUXs to exchange 1 × 8 pixels, along the horizontal direction. In addition, the SBSA has vertical connections to transfer 8 × 1 pixels toward an upper SBSA. The function of the PU is to calculate block SAD, while the function

of the SRU is to buffer search-window pixels for the PU. By the above configuration, a vertical shift operation, left shift operation, right shift operation, and 8 × 8 rectangular-block loading (initial load) are supported, so that a block is searched along the left, right, and lower directions without interruptive pixel reload. The flexible systolic array enables efficient one-dimensional search and FS. The ring-shaped connection is effective for the pixel reuse in the one-dimensional search and FS because the connection does not dispose pixels in search window.

C. Operation of RRSA

The mapping manner of rectangular pixels in the initial loading is illustrated in Fig. 7. A search of consecutive points is executed by shifting. The left, right, and vertical shift operations are depicted in Fig. 8.

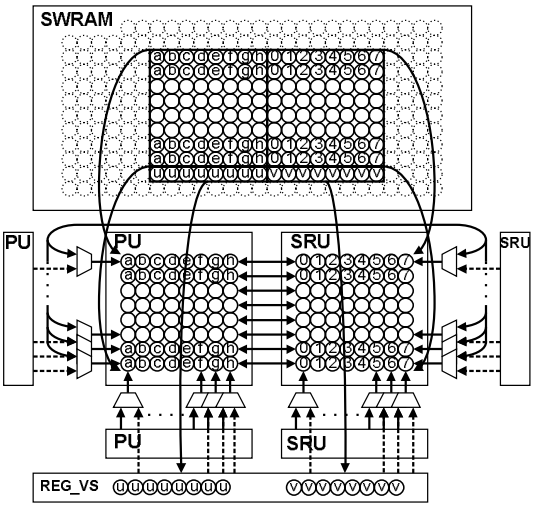

Fig. 7. The mapping manner of rectangular pixels in initial loading.

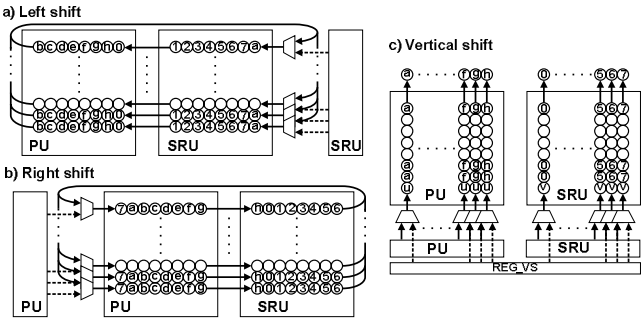

Fig. 8. (a) Left, (b) right, and (c) vertical shifts.

1) One-Dimensional Search

The initial loading to a PU and SRU are required for the one-dimensional search. Only the left shift is employed in the one-dimensional search. Horizontal points are sequentially searched from left to right along the horizontal direction. In contrast, for a one-dimensional search from the top to bottom direction, rectangular pixels are rotated with the cross path circuit before the left shift. In every 8 cycles, 8 × 8 pixels are reloaded to an SRU to search next eight points.

2) Full Search

After initial loading to a PU, SRU and REG_VS, the FS is executed, based on snake search algorithm, which proceeds like along the right, lower, left, and lower directions. The snake search can be achieved with the shift function of the SBSA. The FSs of 8 × 8 and 8 × 16 sizes can be effectively performed in a range of ±4 pixels, with minimal transferred data. Similarly, the FSs of 16 × 8, 16 × 16 and 16 × 32 sizes can be performed in a range of ±8 pixels. The FS is accomplished without a pipeline stall because the pixel data of a next line are supplied while a horizontal shift is performed.

3) One-Time Block Matching on Random Access

As well as the one-dimensional search and FS, one-time block matching can be performed after initial loading to a PU. Because there is no shift operation required, pixel loading to an SRU or REG_VS is needless.

D. Performance of RRSA

Comparison of the required cycles, transferred data and number of transistor between the proposed IME processor and conventional methods are shown in Fig. 9. 512 parallel SIMD architecture and ring-connected systolic array (RCSA) with 512 PEs [6] are evaluated as the conventional method. The SWRAM with accessibility for sub-sampled rectangular 64 pixels is assumed for both the proposed and conventional architectures. The conventional 512-way SIMD architecture requires data transfer from SWRAM every computation cycle so that its transferred data is very large. So it is seen that the SIMD architecture is not suitable for high resolution. The conventional RCSA contains SRU so that the amount of transferred data can be suppressed. However cycle count is still large because it doesn't have the reconfigurable structure so that it is impossible to process smaller-blocks in parallel. The proposed RRSA architecture reduces a cycle count to 28% of the 512-way SIMD and 33% of the 512-way RCSA. The amount of transferred data is reduced to 18 % of the 512-way SIMD. The area of the proposed method is estimated to be increased by 15%, compared to the 512-way RCSA.

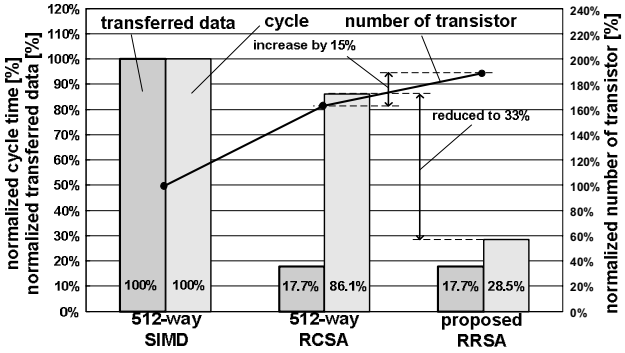

Fig. 9. Execution cycles, transferred data, and transistor number in the proposed RRSA.

IV. Design of SWRAM

A. SWRAM Overview

The proposed SWRAM features one cycle access of a rectangular image data (8 × 8 pixels). A rectangle in an arbitrary location can be accessed without wait-cycle, and the size is variable in a range from 1 × 1 to 8 × 8 pixels. In addition, the SWRAM has a function of horizontal and/or vertical 1/2 sub-sampling. The vertical 1/2 sub-sampling is applied for search in a field picture. Therefore, the SWRAM supplies 8 × 8 pixels in four forms: 8 × 8 rectangle (integer accuracy), 16 × 8 rectangle (horizontal sub-sampling), 8 × 16 rectangle (vertical sub-sampling) and 16 × 16 rectangle (horizontal and vertical sub-sampling). If the above scheme is implemented using the conventional SRAM macro, 256 SRAM-banks are required. In the proposed SWRAM, the number of SRAM banks is reduced to only eight using the merged x-decoder method with the segmentation-free access [2]. The structure and pixel mapping of proposed SWRAM is shown in Fig.10.

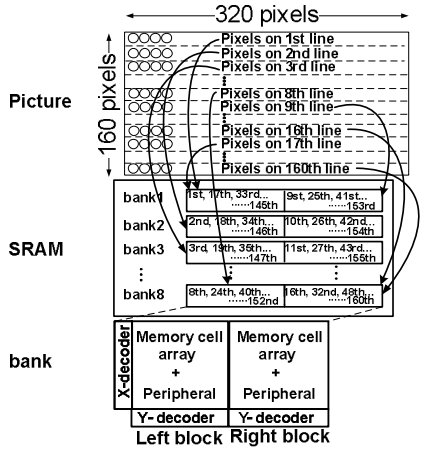

Fig. 10. Structure of SWRAM and pixel-mapping.

B. *Horizontal Segmentation-Free Access*

The segmentation-free accessibility and horizontal sub-sampling is achieved by specific decoders and pixel mapping. Here, we explain the segmentation-free scheme in case of accessing 8 × 1 pixels in a left block. Fig. 11 ("1" signifies the first pixel on a line, and so on) is the schematic of a left block of bank1. With an AND gate, a local word lines are masked by a global word line (GWL) from an X-decoder and local word line select lines (LWLSLs) from a Y-decoder. Pixels on a line are mapped on the block at intervals of 8 pixels. Accessed pixels are selected by switching LWLSL.

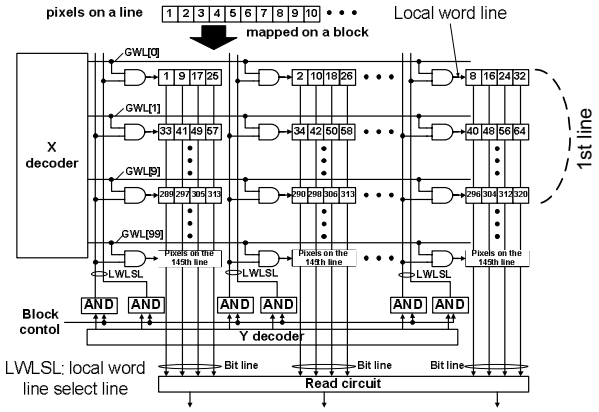

Fig. 11. Schematic of left block in bank1.

C. *Area and Power Estimation of SWRAM*

We compared and evaluated the performances of the proposed and conventional methods in terms of area and power. Here the conventional scheme employs 256 standard SRAM-banks. The area and power of X-decoders turn out to more than 50 % in the conventional, while in the proposed SWRAM they are reduced to less than 5 %. The area and power of the proposed SWRAM as a whole is reduced by 48 % and 49 %, respectively.

V. VLSI IMPLEMENTATION

Fig. 12 is a chip layout of the IME processor core in a 90-nm CMOS technology. A 410-Kb SWRAM corresponds to one reference frame. Multiple reference frames are available by connecting the processors in parallel. The proposed architecture is designed with logic synthesis, while the proposed SWRAM is customized. The processor occupies 2.5 × 2.5 mm^2. The IME core operates at 150 MHz when a nominal supply voltage is 1.0 V. The chip specification is summarized in TABLE I.

The gate-level power estimation accompanied with circuit simulation for memory circuit has been performed. A 100MHz operation supports one-reference-frame search at a resolution of a 30-fps HDTV (1920 × 1080 interlace) with one core, which consumed 48 mW at 1.0 V. In two-core configuration for two reference frame, the power is 96 mW.

TABLE I. CHIP SPECIFICATION

Technology	90nm
Chip size	2.5 x 2.5 mm^2/reference frame
Voltage	1.0 V
Max frequency	150 MHz @ 1.0V
Search range	±128 × ±64
Memory size	SWRAM: 410 kb
Power	48 mW/ref. frame @100MHz 1.0V 96 mW @ 2 reference frames

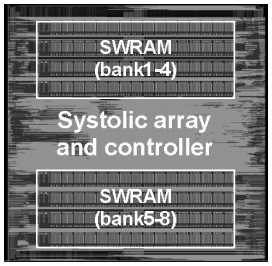

Fig. 12. Chip layout of the proposed IME processor core.

VI. SUMMARY

We mentioned a hierarchical algorithm combining a fine and coarse searches. The fine search is adaptively conducted, based on an image analysis result obtained by the coarse search. The hierarchical algorithm reduces workload by 95 %, and the degradation of picture quality is 0.047 dB. The reconfigurable ring-connected systolic array architecture was implemented to realize the hierarchical algorithm, and minimized the amount of transferred data and computation cycles. The proposed SWRAM supports the rectangular access, segmentation-free access and sub-sampling access, which reduces the power and area to a half, compared to the conventional 256-bank SRAM.

We designed a sub 100-mW H.264/AVC main profile motion estimation processor core for MBAFF encoding. 16 × 16, 16 × 8, 8 × 16, and 8 × 8 block sizes are supported for an HDTV resolution video (1920 × 1080 interlace). The processor provides integer-accuracy motion vectors in real-time operation.

REFERENCES

[1] K. Kumagai "Systen-in-silicon architecture and its application to H.264/AVC motion estimation for 1080HDTV", ISSCC 2006.

[2] S. Warrington "Scalable high-throughput architecture for H.264/AVC variable block size motion estimation", ISCAS 2006.

[3] ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC, Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification, 2003.

[4] J. Miyakoshi, Y. Murachi, T. Ishihara, H. Kawaguchi, and M.Yoshimoto, "A Power- and Area-Efficient SRAM Core Architecture with Segmentation-Free and Horizontal/Vertical Accessibility for Super-Parallel Video Processing," IEICE Trans. Electron., vol.E89-C,no.11, pp.1629-1636, November 2006.

[5] ISO/IEC | ITU-T VCEG, Fast Integer Pel and Fractional Pel Motion Estimation for JVT, JVT-F017, 2002

[6] Y. Murachi, M. Hamano, T. Matsuno, M. Miyakoshi, M. Miyama, and M. Yoshimoto, "A 95mW MPEG2 MP@HL motion estimation processor core for portable high-resolution video application," IEICE Trans. Fundamentals, vol.E88-A, no.12, pp.3492-3499, Dec. 2005.