PAPER *Special Section on Advanced Technologies in Digital LSIs and Memories*

A Sub 100 mW H.264 MP@L4.1 Integer-Pel Motion Estimation Processor Core for MBAFF Encoding with Reconfigurable Ring-Connected Systolic Array and Segmentation-Free, Rectangle-Access Search-Window Buffer

Yuichiro MURACHI[†a)], **Junichi MIYAKOSHI**[†], *Members*, **Masaki HAMAMOTO**[†], **Takahiro IINUMA**[†], **Tomokazu ISHIHARA**[†], **Fang YIN**[†], **Jangchung LEE**[†], **Hiroshi KAWAGUCHI**[†], *Nonmembers*, *and* **Masahiko YOSHIMOTO**[†], *Member*

SUMMARY We describe a sub 100-mW H.264 MP@L4.1 integer-pel motion estimation processor core for 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. The proposed processor features a novel hierarchical algorithm, reconfigurable ring-connected systolic array architecture and segmentation-free, rectangle-access search window buffer. The hierarchical algorithm consists of a fine search and a coarse search. A complementary recursive cross search is newly introduced in the coarse search. The fine search is adaptively carried out, based on an image analysis result obtained by the coarse search. The proposed systolic array architecture minimizes the amount of transferred data, and lowers computation cycles for the coarse and fine searches. In addition, we propose a novel search window buffer SRAM that has instantaneous accessibility to a rectangular area with arbitrary location. The processor core has been designed with a 90 nm CMOS design rule. Core size is $2.5 \times 2.5\,\mathrm{mm}^2$. One core supports one-reference-frame and dissipates 48 mW at 1 V. Two core configuration consumes 96 mW for two-reference-frame search.

key words: *low power, motion estimation, H.264, systolic array, MBAFF, SRAM*

1. Introduction

In H.264, more than double workload of the conventional MPEG2 is necessary for higher picture quality and lower bitrate [1]. The H.264 motion estimation (ME) employs the adaptive block size (ABS) and adaptive frame field (AFF) scheme to achieve higher compression rate, which, however, requires further workload and power. The ABS adaptively handles several block sizes, and improves the picture quality by 0.1 to 0.9 dB. Besides, another 0.5 dB improvement in the picture quality is possible with macro block AFF (MBAFF) that adaptively switches frame/field in every macro block pair (MB-pair). A schema of the ABS and MBAFF is shown in Fig. 1. The workload, however, turns out enormous in these methods because motion vectors (MVs) should be extracted for all combination of the ABS, MBAFF and reference frames.

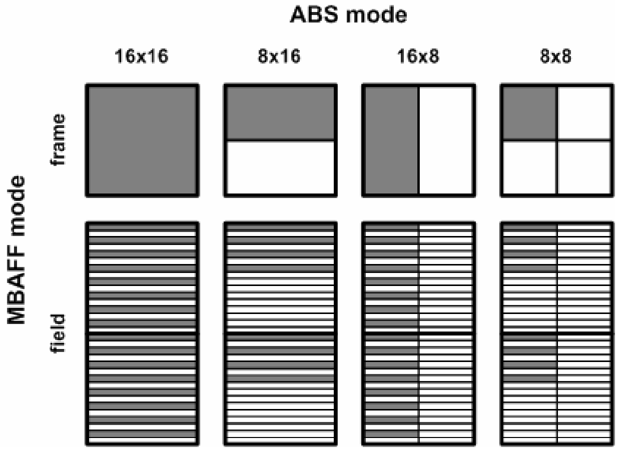


Fig. 1 A schema of the ABS and MBAFF.

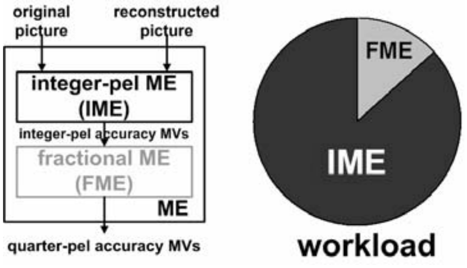


Fig. 2 Workload required for ME.

Figure 2 illustrates a breakdown of workload in the conventional ME. Integer-pel ME (IME) outputs integer-pel accuracy MVs, and then using them, fractional ME (FME) calculates quarter-pel accuracy MVs. According to the workload analysis for baseline profile encoding [2], ME occupies almost 90% of workload in H.264 encoder, and IME occupies about 80% in ME. In case of the main profile with MBAFF option, the workload comes up to several times larger than that of baseline profile with one reference frame because it has to extract motion vectors from three types of reference frame that are frame/field(top)/field(bottom) and B picture needs more than 2 reference frames. So, power consumption in the IME circuit occupies major portion of the entire power of H.264 encoder. This is a motivation of

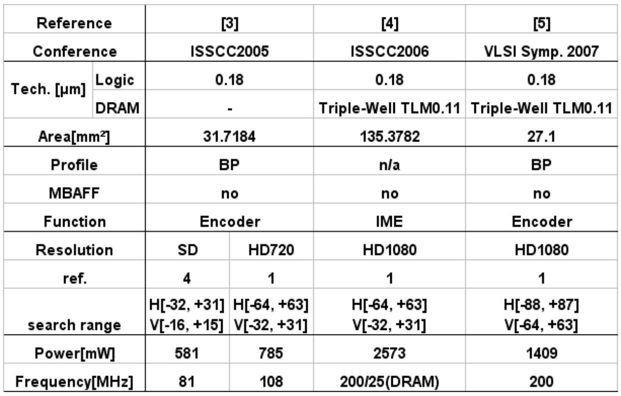
Manuscript received August 18, 2007.
Manuscript revised November 12, 2007.

[†] The authors are with the Kobe University, Kobe-shi, 657-8501 Japan.

a) E-mail: murachi@cs28.cs.kobe-u.ac.jp

DOI: 10.1093/ietele/e91–c.4.465

Table 1 Performance comparison.



Reference		[3]		[4]	[5]
Conference		ISSCC2005		ISSCC2006	VLSI Symp. 2007
Tech. [μm]	Logic	0.18		0.18	0.18
	DRAM	-		Triple-Well TLM0.11	Triple-Well TLM0.11
Area[mm²]		31.7184		135.3782	27.1
Profile		BP		n/a	BP
MBAFF		no		no	no
Function		Encoder		IME	Encoder
Resolution		SD	HD720	HD1080	HD1080
ref.		4	1	1	1
search range		H[-32, +31] V[-16, +15]	H[-64, +63] V[-32, +31]	H[-64, +63] V[-32, +31]	H[-88, +87] V[-64, +63]
Power[mW]		581	785	2573	1409
Frequency[MHz]		81	108	200/25(DRAM)	200

research to reduce a power of IME.

Several H.264 encoders and ME circuits have been presented ([3]–[5]). Table 1 summarizes the power, area, and other specifications of these existing solutions. Their designs were limited only for the baseline profile and they did not support the main profile and MBAFF mode which realizes higher quality characteristics. In order to implement the main profile with MBAFF mode, low power design solutions for its complicated processing and the enormous computation load are desirable.

This paper proposes a H.264 MP@L4.1 IME processor core solution to overcome the above problem and to achieve the main profile and MBAFF encoding. Here mutual optimizations among algorithm, architecture and circuit design layer have been executed to achieve a low power realization. We propose the following three techniques to realize MBAFF, allowing high picture quality and low power characteristic at the same time:

- An IME algorithm which adopts hierarchical and adaptive search methods with image analysis for MBAFF coding.
- A reconfigurable ring-connected systolic array architecture which minimizes the amount of transferred pixel data and lowers computational cycles with the search methods.
- A search window buffer SRAM which supports instantaneous rectangular access, segmentation-free access and sub-sampling access with small silicon and at low power consumption.

The proposed IME algorithm is hierarchical search method with a combination of the ABS and MBAFF. UMHexagon search [6] which is the conventional IME search method, abundantly makes early termination which reduces an average workload. But UMHS is not suitable for VLSI implementation because of its large workload in the worst case. Instead, the 1-dimensional diamond search (1D-DS) [7] reduces the worst-case workload, although, in the HDTV resolution, picture quality is deteriorated. Meanwhile, the proposed algorithm adopts a hierarchical search method comprised of a coarse search and fine search. After the coarse search, the calculated MVs are analyzed whether directions and magnitudes of the MVs are similar or differ largely. Then, the fine search is adaptively carried out, according to the analysis. By this algorithm, both high picture quality and light worst-case workload are achieved.

The proposed architecture is a reconfigurable ring-connected systolic array architecture which provides the minimal amount of transferred pixel data and low computational cycles. Most of the conventional architectures [8], [9] are dedicated only to one search method, so the conventional ones are not suitable for adaptive search algorithms. On the other hand, our architecture supports three types of search methods: a full search (FS), 1-dimensional search and random block matching with eight sub-block systolic arrays. Furthermore, the proposed architecture supports all block sizes, the MBAFF modes, and sub-sampling search modes with a combination of sub-block systolic arrays.

Note that it is necessary to supply an arbitrary rectangular area from a search window RAM (SWRAM). An SRAM with 256 banks is required to fulfill the demand, which, however, would result in a large area and power in decoder circuits. To resolve the issue, we propose a novel SWRAM which has a segmentation-free accessibility. Rectangular access to an arbitrary location is also supported in the proposed SWRAM at low power.

In this paper, the detail of the proposed IME algorithm is described in Sect. 2. The proposed reconfigurable ring-connected systolic array architecture is addressed in Sect. 3. Section 4 explains the architecture of proposed SWRAM. Then, these are followed by VLSI implementation and power estimation in Sect. 5. Section 6 concludes this paper.

2. Algorithm

2.1 Algorithm Overview

The proposed algorithm is a hierarchical search algorithm comprised of a coarse search and fine search. The fine search is adaptively carried out, according to an analyzing result of the coarse search. The whole algorithm flow is shown in Fig. 3. 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 search method with low complexity and high picture quality, named complementally recursive cross search (CRCS). According to the analysis result, the flow is branched to two types of fine search.

One is the fine search by smaller-block (FSSB) which can detect globally-distributed MVs with smaller block. In the other case, a fine search by MB-pair (FSMP) is chosen to find locally-distributed MVs with fewer workload. By using either of the fine search methods, all combination of two MBAFF modes and four block sizes (i.e. field-16 × 16, field-16 × 8, field-8 × 16, field-8 × 8, frame-16 × 16, frame-16 × 8, frame-8 × 16, frame-8 × 8) are carried out. In this section, the coarse search, image analysis, and the two kinds of fine searches are described in Sects. 2.2, 2.3, and 2.4, respectively. These are followed by simulation results of the proposed IME algorithm in Sect. 2.5.

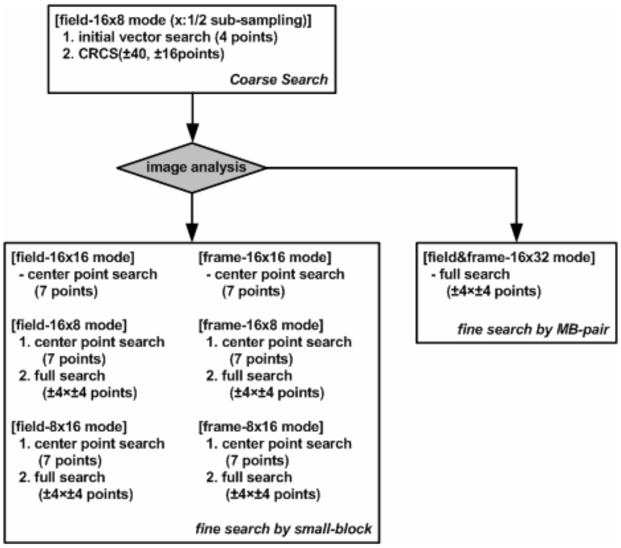

[field-16x8 mode (x:1/2 sub-sampling)]
1. initial vector search (4 points)
2. CRCS(±40, ±16points)
Coarse Search
image analysis
[field-16x16 mode]
- center point search (7 points)
[frame-16x16 mode]
- center point search (7 points)
[field-16x8 mode]
1. center point search (7 points)
2. full search (±4×±4 points)
[frame-16x8 mode]
1. center point search (7 points)
2. full search (±4×±4 points)
[field-8x16 mode]
1. center point search (7 points)
2. full search (±4×±4 points)
[frame-8x16 mode]
1. center point search (7 points)
2. full search (±4×±4 points)
fine search by small-block
[field&frame-16x32 mode]
- full search (±4×±4 points)
fine search by MB-pair

Fig. 3 IME algorithm flow.

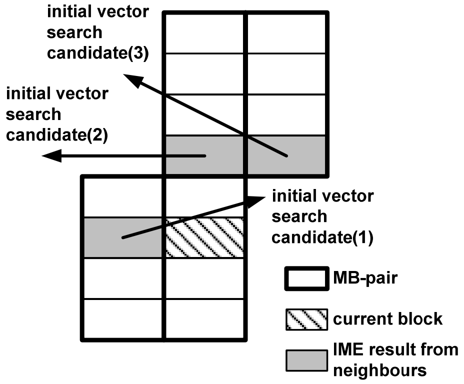

initial vector search candidate(3)
initial vector search candidate(2)
initial vector search candidate(1)
MB-pair
current block
IME result from neighbours

Fig. 4 Candidate MVs from neighbor MB-Pairs.

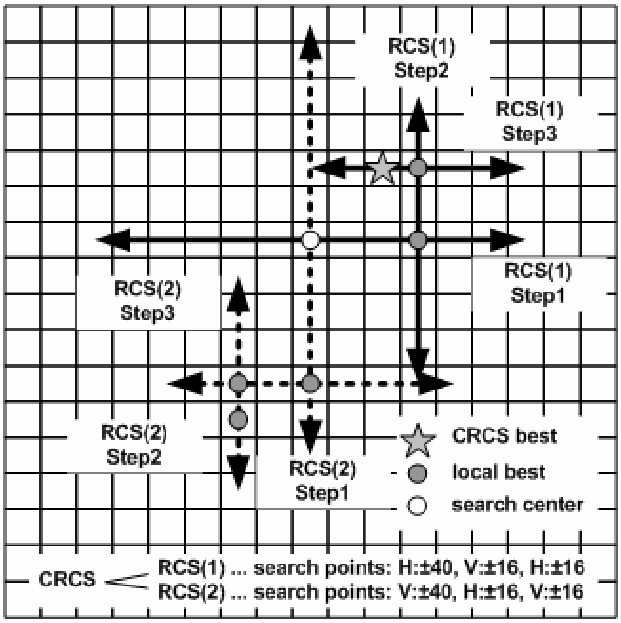


Fig. 5 Complementally RCS (CRCS).

2.2 Coarse Search

In the coarse search, two searches called an initial vector search and the CRCS are performed sequentially on a subsampling image. To obtain the suboptimal MVs in a wide search range, one MV out of four candidate MVs are chosen in the initial vector search, and then the CRCS is executed.

2.2.1 Initial Vector Search

The following four candidate MVs are evaluated in the initial vector search:

- (0, 0) vector (1 point)
- The best IME results from neighboring blocks (3 points)

It is well known that a predictive MV (PMV) is effective as a candidate vector to enhance picture quality. However it is difficult to implement PMV on VLSI architecture because PMV is calculated using result of FME, which caused pipeline stall in pipelined hardware. So we adopt the best search result of the IME from a left, upper and upper right MB-pair as the three candidates, instead of the PMV. The candidate vectors are illustrated in Fig. 4. By the initial vector search, one of 4 candidates is chosen to be utilized as a starting vector in the following CRCS.

2.2.2 Complementally Recursive Cross Search (CRCS)

In this subsection, we mentions the recursive cross search (RCS) and CRCS that are based on a gradient search method with enhancing parallelism. One of the conventional gradient search methods is the conjugate direction search (CDS) [10]. CDS can reduce an average workload with less quality degradation. The algorithm is, however, not suitable for VLSI implementation because many conditional branch operations induce frequent pipeline stalls, thus increases number of cycles. Also diagonal search direction prevents pixel reusing between neighboring search points, which makes power and data rate reduction difficult. To solve the above problems, a new gradient search algorithm (RCS) is proposed. The RCS is illustrated in Fig. 5. The operation of RCS algorithm is described as follows. Firstly, ±40 points in the horizontal direction are evaluated by 1-dimensional search (1-DS) which detects MVs in continuous points on a straight line. The center of the ±40 points at Step 1 is the result of the initial vector search. Secondly, the search continues over ±16 points in the vertical direction, taking the point with minimal SAD in step1 as search center (step2). By this way, horizontal ±16 points whose center is result of step2 are searched (step3). The result of RCS is the point with smallest SAD among all search points.

Whether the higher picture quality is given by RCS start from horizontal direction or vertical direction depends on characteristics of an image sequence. Therefore, CRCS method is employed to improve picture quality by comple-

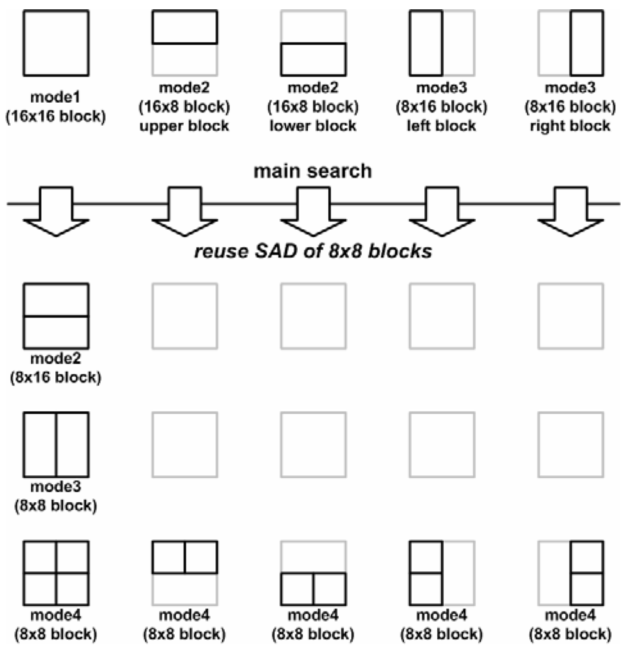


Fig. 6 SAD reusing.

mentarily using 2 RCSs starting with the horizontal direction search and vertical direction search respectively. In CRCS, 2 RCSs 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.

CRCS is performed using block type of field-16 × 8 horizontally sub-sampled. "Horizontally sub-sampled" means that both the points of CRCS and pixels of SAD are sub-sampled by a factor of 2, and that it can reduce workload effectively. As shown in Fig. 6, smaller block SADs on the same point are obtained by a search of larger block such as 16 × 16 with reuse of partial SADs. So SAD of field-8 × 8 horizontally sub-sampled is given by coarse search with SAD reuse. The horizontal sub-sampling field-16 × 8 is equivalent to the horizontal and vertical sub-sampling frame-16 × 16. Horizontal and vertical sub-sampling SADs of frame-16 × 8, frame-8 × 16 and frame-8 × 8 are obtained from the horizontal and vertical sub-sampling SADs of frame-16 × 16 by coarse search with SAD reuse. Consequently, field-16 × 8, field-8 × 8, frame 16 × 16, frame-16 × 8, frame-8 × 16 and frame-8 × 8 are evaluated in the coarse search. The best MV obtained by coarse search is suitable for the image analysis, because it has the feature of 6 block-types.

2.3 Image Analysis

By using integer-pel MV (Fig. 7) in field-16 × 8 of coarse search, distribution of MVs is analyzed to reduce workload. If both the temporal condition shown in expression (1), (2) and the spatial condition in expression (3), (4), (5), (6) are under the threshold value (= 4), analysis result means that MVs are locally-distributed, and search algorithm is branched to FS by MB-pair (FSMP) which result in low workload. If any of Eqs. (1)–(6) is not satisfied, analysis result means that MVs are globally distributed and the algorithm is branched to fine search by smaller-block (FSSB).

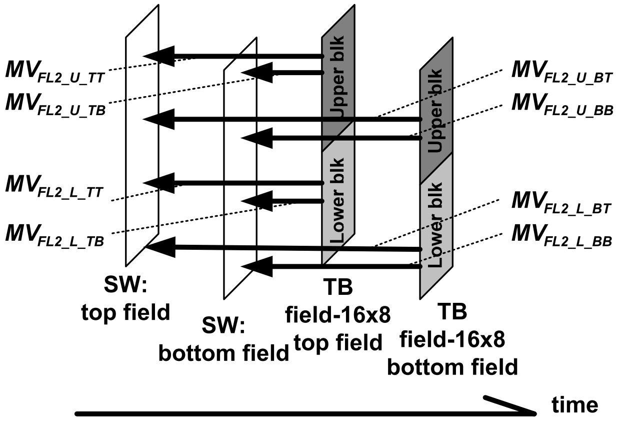


Fig. 7 Image analysis using MVs obtained from coarse search.

Temporal condition:

$$|MV_{FL2\_U\_TT} - MV_{FL2\_U\_BB}| < THR_{PATH} \tag{1}$$

$$|MV_{FL2\_L\_TT} - MV_{FL2\_L\_BB}| < THR_{PATH} \tag{2}$$

Spatial condition:

$$|MV_{FL2\_U\_TT} - MV_{FL2\_L\_TT}| < THR_{PATH} \tag{3}$$

$$|MV_{FL2\_U\_TB} - MV_{FL2\_L\_TB}| < THR_{PATH} \tag{4}$$

$$|MV_{FL2\_U\_BT} - MV_{FL2\_L\_BT}| < THR_{PATH} \tag{5}$$

$$|MV_{FL2\_U\_BB} - MV_{FL2\_L\_BB}| < THR_{PATH} \tag{6}$$

("| |" means summation of x element and y element.)

2.4 Fine Search

2.4.1 Fine Search by Smaller-Block (FSSB)

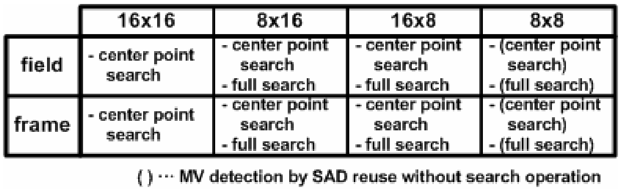
In FSSB, the search of optimum points for each block-type is performed individually. For each block of 16 × 16, 8 × 16 and 16 × 8 in both frame and field mode, center point search (CPS) is performed to detect the center point for the succeeding FS. A CPS is performed by random block matching (RBM) method with following 7 candidate search points:

- (0, 0) vector (1 point).
- The best IME results from neighboring blocks (3 points).
- The best result from coarse search (3 points).

The picture quality is improved by performing individual FS (search range: ±4 × ±4) for 16 × 8 and 8 × 16 block. As shown in Fig. 8, SAD of 8×8 block is obtained by reusing SAD from block size 16 × 8 and 8 × 16.

2.4.2 Fine Search by MB-Pair (FSMP)

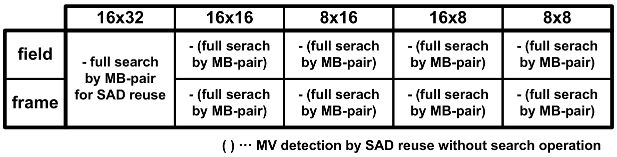
In the case that MVs of small block are locally-distributed, FS (search range: ±4 × ±4) by MB-pair (FSMP) is performed. As shown in Fig. 9, MVs for each block type of



	16x16	8x16	16x8	8x8
field	- center point search	- center point search - full search	- center point search - full search	- (center point search) - (full search)
frame	- center point search	- center point search - full search	- center point search - full search	- (center point search) - (full search)

( ) ··· MV detection by SAD reuse without search operation

Fig. 8 Fine search by small-block covers all mode.



	16x32	16x16	8x16	16x8	8x8
field	- full search by MB-pair for SAD reuse	- (full serach by MB-pair)	- (full serach by MB-pair)	- (full serach by MB-pair)	- (full serach by MB-pair)
frame		- (full serach by MB-pair)	- (full serach by MB-pair)	- (full serach by MB-pair)	- (full serach by MB-pair)

( ) ··· MV detection by SAD reuse without search operation

Fig. 9 Fine search by MB-pair covers all mode.

Table 2 Simulation condition.

Profile	main profile
Resolution	HDTV (1920 x 1080 interlace) SDTV (720 x 480 interlace)
Frame rate	30 fps
# reference pictures	bidirectional 2 frames
adaptive frame field (AFF)	macro block AFF
Others	No R-D optimization No rate control
Sequence	bronze with credit, church, european market, whale jump, soccer action
# frames	30
Search range	HDTV : ±128 × ±64 SDTV : ±64 × ±32

16 × 16, 8 × 16, 16 × 8 and 8 × 8 in both frame and field mode are extracted by reusing SADs obtained from FS by MB-pair, with lower workload keeping high picture quality.

2.5 Simulation Result

Table 2 shows the simulation condition. QP is set to the value providing 10 Mbps in HDTV resolution and 2.5 Mbps in SD resolution on bitrate and 4 point around it. Block size used on the simulation are 16 × 16, 16 × 8, 8 × 16 and 8 × 8, and MBAFF mode is utilized. The 2-step search with surrounding 8 point is employed as FME algorithm. Conventional methods for comparison are FS which has a search range of ±128 × ±64 and UMHS which is adopted in the H.264 joint model (JM) [11]. Figure 10 shows test sequences used in simulation.

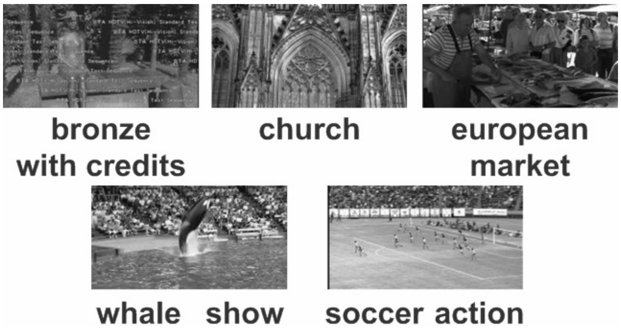


Fig. 10 Test sequences.

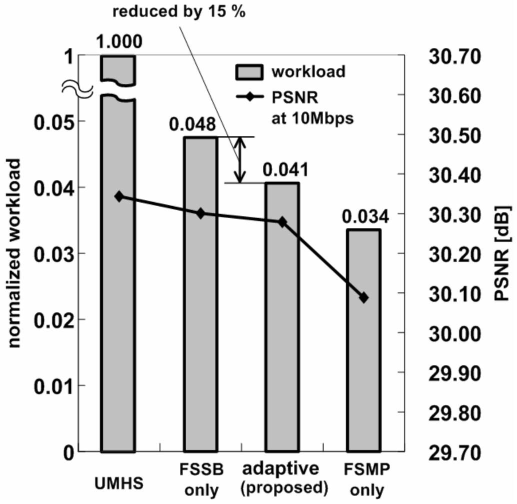


Fig. 11 Simulation result of adaptive search in HDTV resolution.

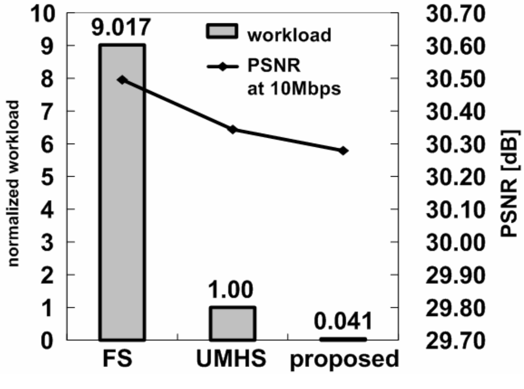


Fig. 12 Comparison of workload and picture quality in HDTV resolution between the proposed and conventional algorithms.

Normalized workload and PSNR in HDTV resolution are summarized in Fig. 11. By using adaptive fine search, workload is reduced by 15% compared to fine search by smaller-block (FSSB) with little degradation of picture quality. Figure 12 shows comparison of workload and picture quality in HDTV resolution between the proposed and conventional algorithms. Workload of the proposed algorithm is reduced to 4.1% and picture degradation is only 0.06 dB in comparison with UMHS. Meanwhile a simulation in SD resolution revealed that the workload was reduced to 9.1% and picture degradation was only 0.03 dB in comparison with UMHS. In case of 3 reference frames in SD resolution, picture quality was enhanced by about 0.07 dB for both UMHS and the proposed algorithm.

3. Architecture

The search methods in the proposed IME algorithm are the FS, CRCS, and RBM. In this section, a reconfigurable ring-connected systolic array (RRSA) is proposed, which supports the above three search methods efficiently at less data transfer and in fewer cycles. It contains 8 systolic arrays, each for sub-block (8 × 8 block). The interconnections among 8 systolic arrays are reconfigured to handle multiple block sizes and the two MBAFF modes. Moreover, it has a structure to allow the reuse of calculated SAD, so that a plural motion vectors in different block type and different mode can be obtained simultaneously.

3.1 Architecture Overview

Figure 13 shows a whole block diagram of the proposed IME processor. The proposed IME processor consists of an RRSA, an SWRAM (2 ports: 1 read/1 write) that is search window buffer and provides reference image data for the RRSA, a cross path circuit that rotates pixel data, a template-block buffer (TB buffer: register file) that provides current image data, and a controller. The IME processor has a novel RRSA architecture which is comprised of eight systolic arrays called sub-block systolic arrays (SBSAs). An SBSA calculates SAD for 8 × 8 size block. The eight SBSAs can be compound to 8×8 to 32×16 systolic arrays to support the all modes of the ABS and MBAFF in H.264. The operation of RRSA is programmable and three search methods of the FS, 1-DS, and RBM are supported.

To execute the adaptive search using the RRSA architecture, functions needed for SWRAM are to supply rectangular and sub-sampling pixels for the hierarchical search. Various forms of rectangular pixel accesses are needed in the H.264 encoding, shown in Fig. 14. A 8 × 8 rectangle image (integer-pels), 16 × 8 rectangle image (horizontal sub-sampling), 8 × 16 rectangle image (vertical sub-sampling), and 16 × 16 rectangle image (horizontal and vertical sub-sampling) have to be accessed from the search window buffer within one cycle. For this purpose, the novel low-power small-area SWRAM architecture as the search window buffer is proposed. The SWRAM is introduced in Sect. 4. The register for a vertical shifter (REG_VS) and the cross path extend flexibility by buffering and rotating pixels from the SWRAM. Figure 15 shows the rotation of pixels, as an example.

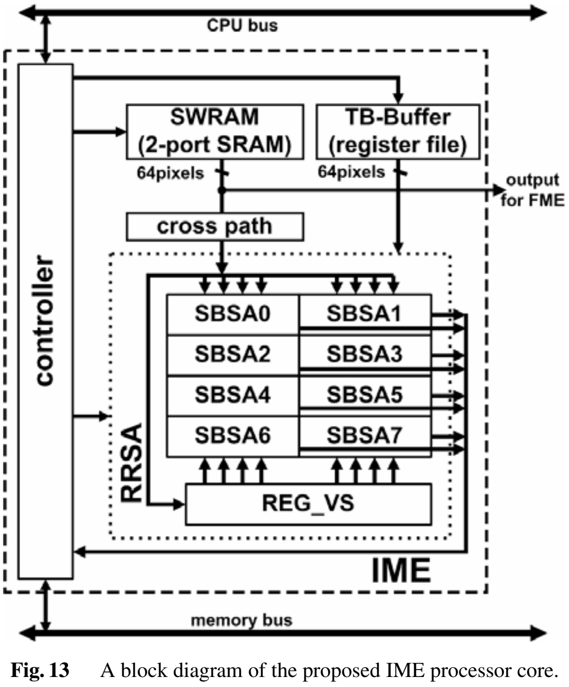


Fig. 13 A block diagram of the proposed IME processor core.

3.2 Reconfigurable Ring-Connected Systolic Array

Figure 16 shows a block diagram of the RRSA. The RRSA

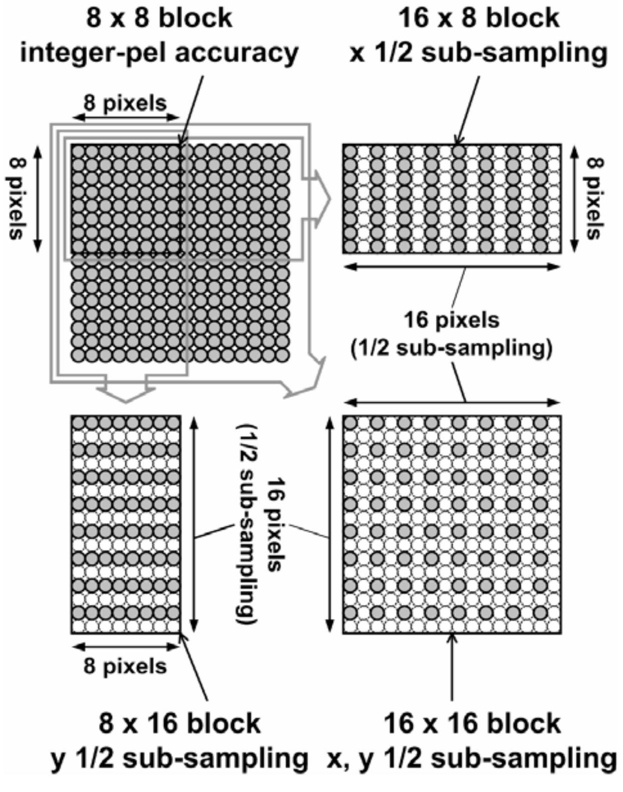


Fig. 14 Required pixels in sub-sampling.

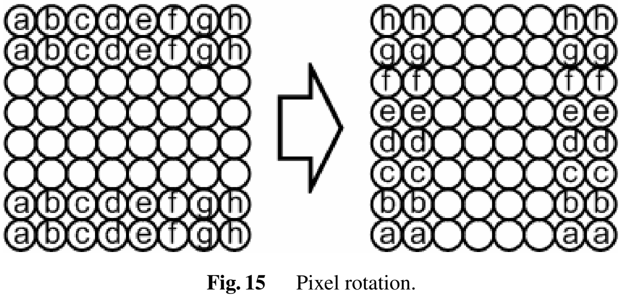


Fig. 15 Pixel rotation.

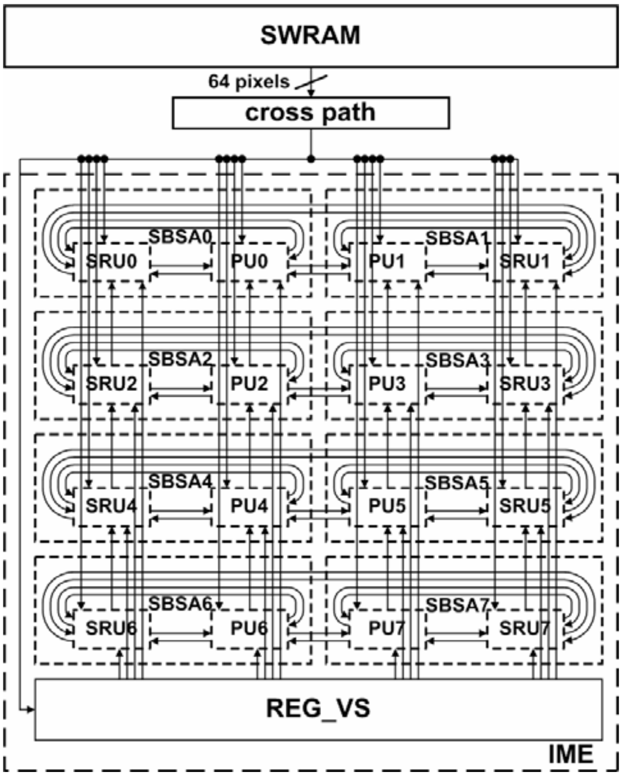


Fig. 16 A block diagram of RRSA.

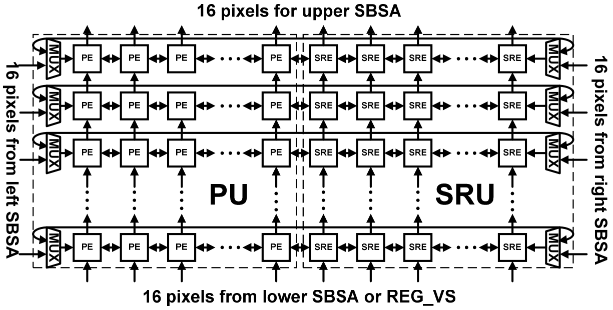


Fig. 17 A block diagram of SBSA.

consists of eight SBSAs and one REG_VS as a vertical pixel buffer. A SBSA consists of a 8 × 8 processing unit (PU) and 8 × 8 shift register unit (SRU), which provides SAD calculation of a sub block (8 × 8 block size). SBSAs are connected with adjacent SBSAs by bi-directional ring interconnections. Horizontally- and vertically-connected SBSAs are grouped to calculate SAD of larger blocks (16 × 8, 8 × 16, 16 × 16, 16 × 32 block sizes).

3.2.1 SBSA Architecture

Figure 17 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.

Figure 18 shows a block diagrams of a PE and SRE. One pixel from SWRAM and 1 pixel from TB buffer are stored in a PE, then absolute difference between the SW pixel and TB pixel is calculated.

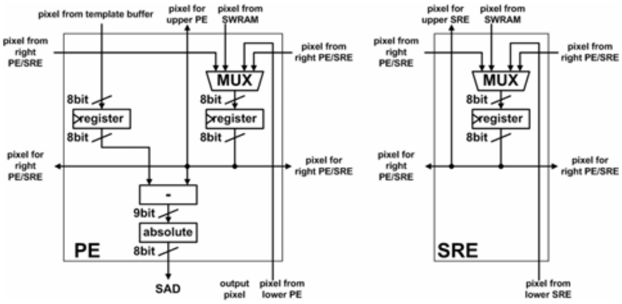


Fig. 18 A processor element and a shift register.

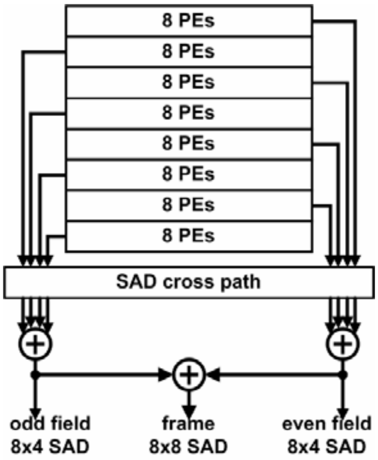


Fig. 19 Adder tree of SAD.

As shown in Fig. 19, a PU calculates 8 × 4 SAD for the field mode (odd field), 8 × 4 SAD for the field mode (even field) and 8 × 8 SAD for the frame mode. By changing a combination of SBSAs, PE arrays from an 8 × 8 to 32 × 16 can be configured. Figure 20 shows the combinations of SBSAs. 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 RRSA architecture.

The PU and SRU have a direct access path for initial load and buffered access path for FS. Furthermore the PU and SRU allows to search at continuous points without interruptive reload of pixels from the SWRAM. The SBSA has an inter and inner ring-connection to exchange 1 × 8 pixels with a horizontally-connected SBSA. The SBSA also has a vertical connection to transfer 8 × 1 pixels towards an upper SBSA. The function of the PU is to calculate block SADs, while the function of the SRU is to buffer search-window pixels for the PU. By this mechanism, a vertical shift operation, left shift operation, right shift operation, and 8 × 8 rectangular block load (initial load) are supported by the SBSA, so that continuous points are searched towards left, right, and lower directions without interruptive pixel

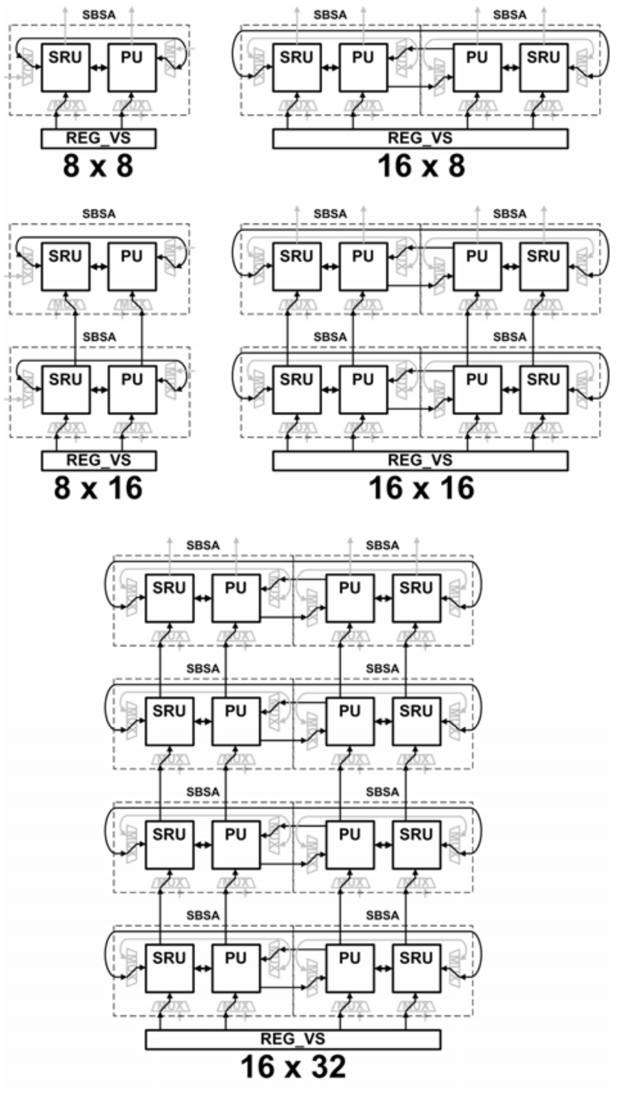


Fig. 20 Combinations of SBSAs.

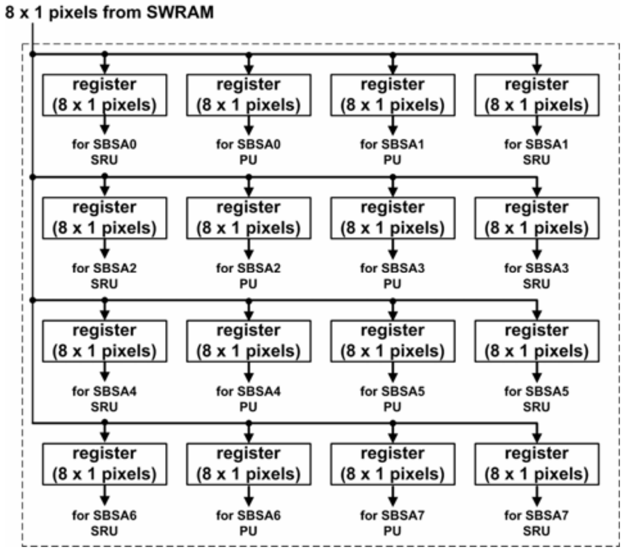


Fig. 21 A block diagram of REG_VS.

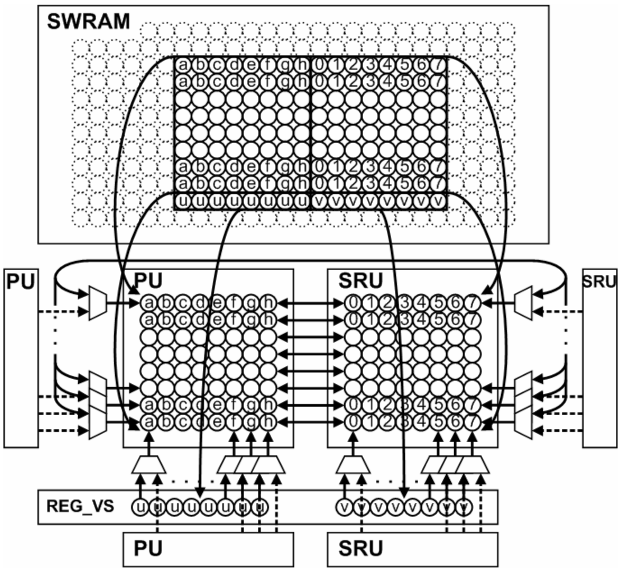


Fig. 22 The mapping manner of rectangular pixels in the initial load.

reload. These continuous shifts enable efficient 1-DS and FS search methods. The ring-shaped connection is effective for the pixel reuse in the 1-DS and FS because the connection prevents to dispose search-window pixels.

3.2.2 REG_VS Architecture

The vertical shift in the FS operation is supported by delay line registers in REG_VS. Figure 21 shows a block diagram of REG_VS. Each SBSA has 16 × 1 registers in REG_VS for its own use, so REG_VS has 8 × 16 × 1 pixel registers. REG_VS receives 16 × 1 pixels (taking 2 cycles) from the SWRAM and stores them for the vertical shift operation in an SBSA. When a searching point of the FS moves down, pixels in the SBSAs shift above. Pixels from REG_VS are transferred to all SBSAs from below in one cycle.

3.3 Operation of RRSA

The three search methods such as 1-DS, FS, and RBM are realized by shift operations after initial data loading in the RRSA. The mapping manner of rectangular pixels in the initial load is shown in Fig. 22. If a rotation of the block is required, pixels are mapped in the manner shown in Fig. 23. A search of consecution points are executed by shifting. The operations of left shift, right shift and vertical shift are shown in Fig. 24.

3.3.1 1-Dimensional Search Operation

The initial loads for a PU and SRU are required for the 1-

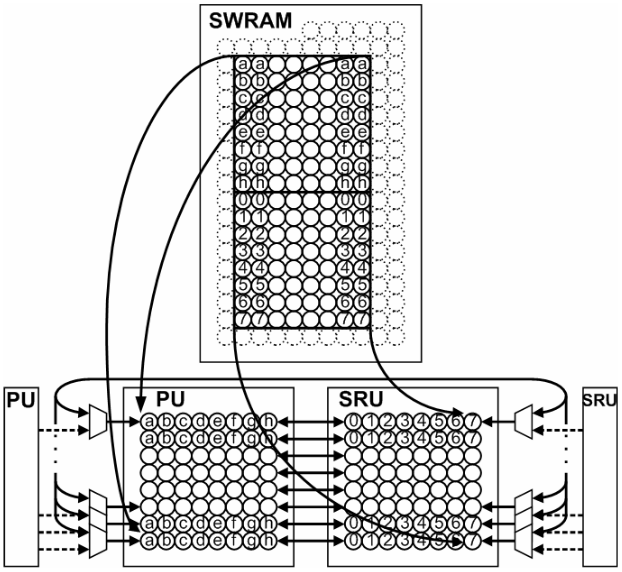


Fig. 23 The mapping manner of rectangular pixels in the initial load in the rotating case.

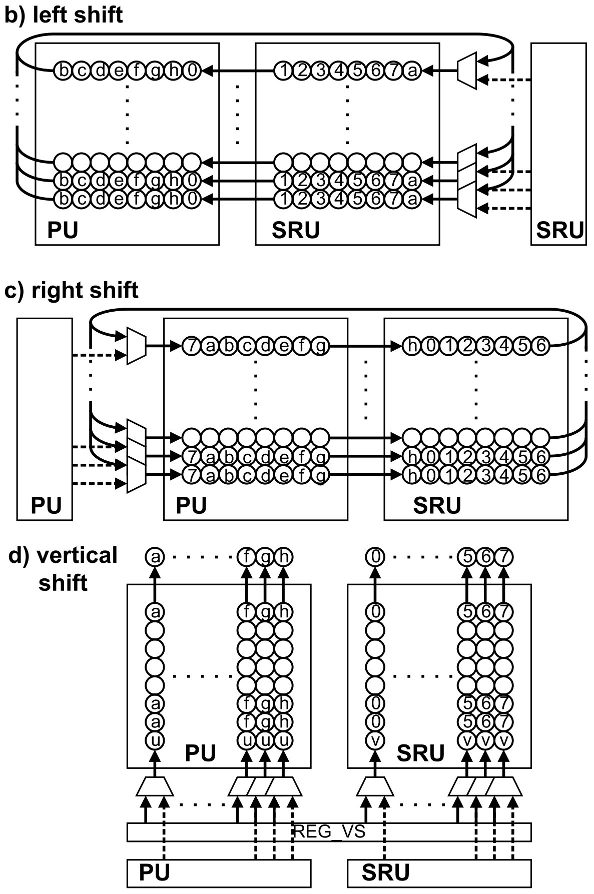


Fig. 24 Detailed views of shift operations.

DS operation. Only left shift is employed in 1-DS operation. Horizontal points are sequentially searched from left to right in the horizontal 1-DS, also vertical points are sequentially searched from top to bottom in the vertical 1-DS, with the rotation of the block followed by left shift. Every 8 cycles, 8 × 8 pixels are initially loaded into an SRU.

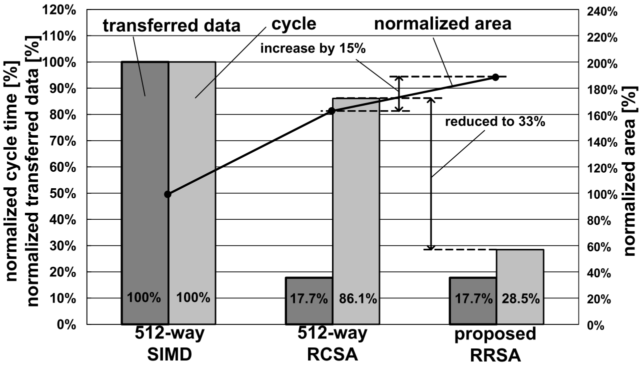


Fig. 25 Comparison of the required cycles, transferred data and area between the proposed RRSA and conventional methods.

3.3.2 Full Search Operation

The initial loads for a PU, SRU and REG_VS are required in the FS operation, as well. The FS is executed as a snake search, which is performed in orders of right, below, left, below, and so on. This snake search is realized in the SBSA, by operations of a left shift, vertical shift, right shift, and vertical shift, and so on. The FSs of 8 × 8 and 8 × 16 size can be performed effectively in a range less than ±4, and the FSs of 16 × 8, 16 × 16 and 16 × 32 sizes can be performed in a range less than ±8 with minimal transferred data. Because the pixel data of a next line can be supplied while a horizontal shift is performed, the FS is accomplished without pipeline stall.

3.3.3 RBM Operation

In RBM, an initial load for a PU is required. Because there is no shift operation in RBM, a pixel load for an SRU or REG_VS is not needed.

3.4 Performance of RRSA

Comparison of the required cycles, transferred data and area between the proposed IME processor and conventional methods are shown in Fig. 25. In this picture, area is evaluated after logic synthesis and layout. 512 parallel SIMD architecture and ring-connected systolic array (RCSA) with 512PEs [7] are evaluated as the conventional method. The SWRAM providing rectangular and sub-sampling 64 pixels accessibility 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 ring-connected systolic array (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 reconfigurability 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.

4. SWRAM

The proposed SWRAM features one cycle access of a rectangular image data (m×n 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. In addition, the SWRAM has a function of horizontal and/or vertical 1/2 sub-sampling. 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 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 a merged x-decoder method with the segmentation-free access [12].

4.1 SWRAM Overview

A block diagram of the SWRAM is shown in Fig. 26. The figure also shows a mapping manner of pixels in a search window. The SWRAM consists of eight banks: each has left and right blocks. Pixels on the search window are mapped on the SWRAM line by line so that consecutive lines can be accessed parallel in a cycle. The value of M is positive integer. Each block also has segmentation-free accessibility [12] by which 8 (integer accuracy) or 16 (horizontal sub-sampling) consecutive points in an arbitrary location can be accessed.

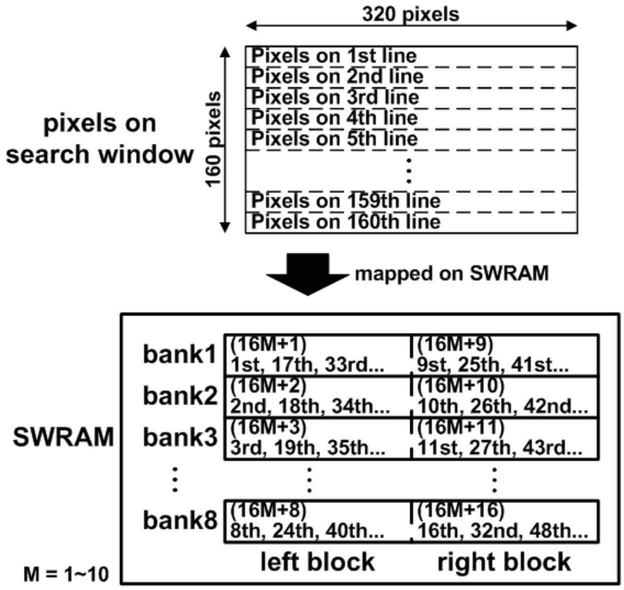


Fig. 26 A block diagram of SWRAM and its pixel mapping.

4.2 Bank Configuration

A block diagram of an SRAM bank is shown in Fig. 27. X-decoders of the left block and the right block are merged to reduce the area and power. Left and right blocks are accessed independently by with an aid of AND circuits which is switched by the block control signal.

4.3 Segmentation-Free Access in X-Direction

Segmentation-free accessibility and horizontal sub-sampling is realized by a unique connection between decoders and specific mapping manner of pixels. Here the segmentation-free scheme is explained using the left block. Block diagram of the left block and mapping manner of pixels on a line are shown in Fig. 28. Logical value of local word lines

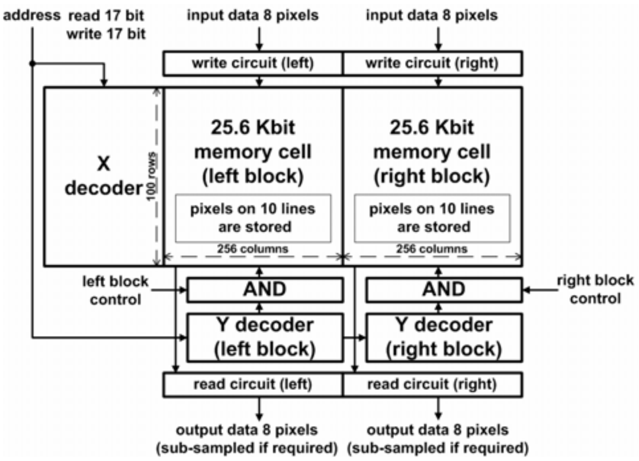


Fig. 27 SRAM bank.

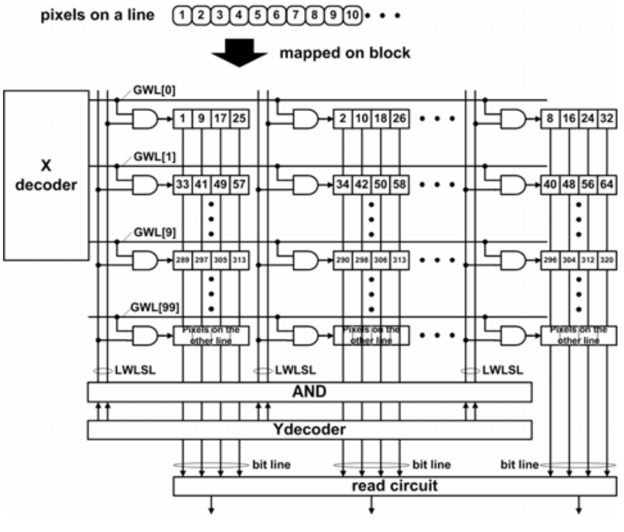


Fig. 28 Mapping manner. “1” signifies the first pixel on a line, and so on.

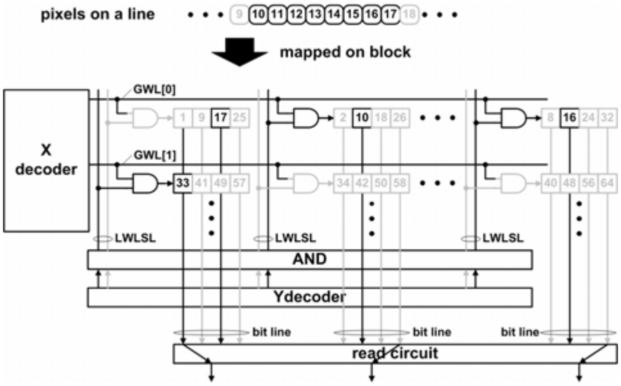


Fig. 29 Segmentation-free access to 8 pixels (integer accuracy).

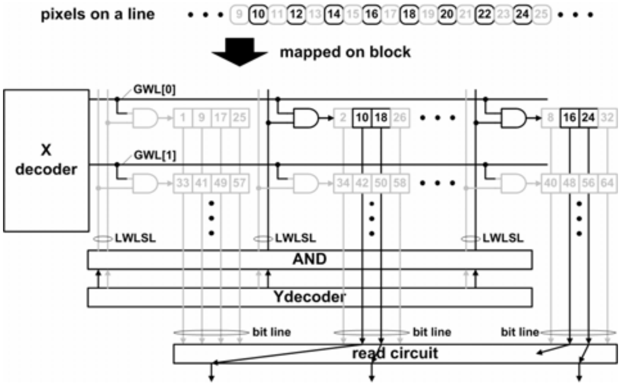


Fig. 30 Segmentation-free access to16 pixels (horizontal sub-sampling).

(LWLs) are decided by AND operation between global word line from X-decoder and LWL select lines (LWLSLs) from Y-decoder. Pixel on a line is mapped on the block 8 pixels by 8 pixels. Accessed pixels are selected by switching 16 (2 × 8 lines) values of LWLSL. Figure 29 and Fig. 30 show operation of 8 pixels (integer accuracy) access and 16 pixels (1/2 sub-sampling) access, respectively. In these figures, black lines mean signals in active. 16 X-decoders with conventional SRAM scheme are reduced to 1 X-decoder in proposed SRAM block.

4.4 Rectangular Accessibility

The SWRAM provides the rectangular accessibility and the segmentation-free accessibility. All of 8 × 8 pixels in 4 forms that are 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), are fetched in a single cycle from SWRAM. Figure 31 shows 8×8 rectangular access (integer accuracy) and Fig. 32 shows 16×16 rectangular access (horizontal and vertical sub-sampling). Functions of the vertical consecutive 8 lines access, the vertical segmentation free access and the vertical sub-sampling are supported by combination of 16 SRAM blocks, while horizontal functions are supported by each SRAM block.

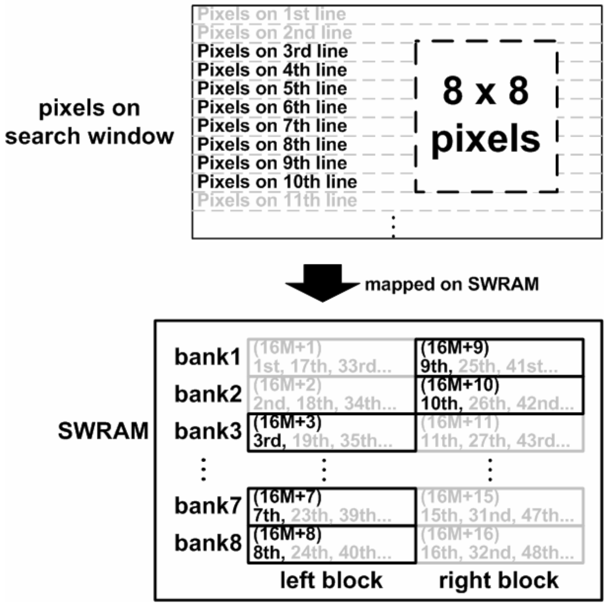


Fig. 31 Access operation of 8 × 8 pixels.

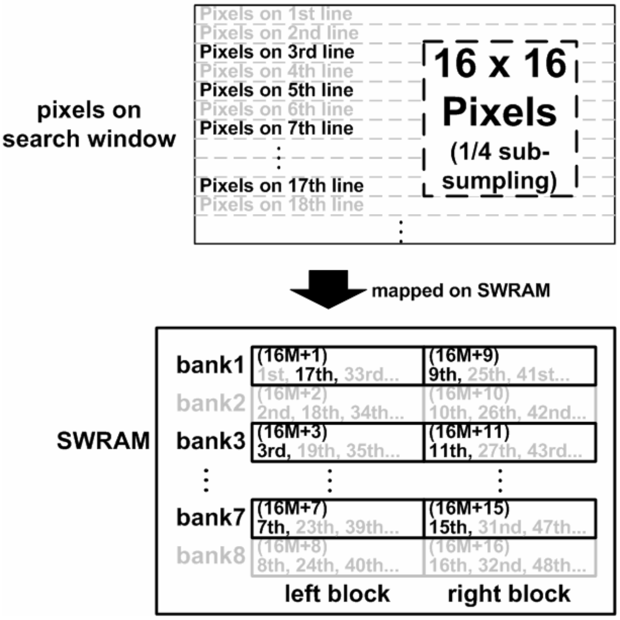

pixels on search window
Pixels on 1st line
Pixels on 2nd line
Pixels on 3rd line
Pixels on 4th line
Pixels on 5th line
Pixels on 6th line
Pixels on 7th line
Pixels on 17th line
Pixels on 18th line
16 x 16 Pixels (1/4 sub-sampling)
mapped on SWRAM
SWRAM
bank1
bank2
bank3
bank7
bank8
(16M+1) 1st, 17th, 33rd...
(16M+2) 2nd, 18th, 34th...
(16M+3) 3rd, 19th, 35th...
(16M+7) 7th, 23th, 39th...
(16M+8) 8th, 24th, 40th...
(16M+9) 9th, 25th, 41st...
(16M+10) 10th, 26th, 42nd...
(16M+11) 11th, 27th, 43rd...
(16M+15) 15th, 31nd, 47th...
(16M+16) 16th, 32nd, 48th...
left block
right block

Fig. 32 Access operation of 16 × 16 pixels (horizontal and vertical sub-sampling).

4.5 Area and Power Estimation of SWRAM

Performances of the proposed and 2 conventional methods are compared and evaluated. The following 2 schemes are examples of conventional methods to provide rectangular pixels accessibility in 4 forms — 8 × 8, 16 × 8 (horizontal sub-sampling), 8 × 16 (vertical sub-sampling), 16 × 16 (horizontal and vertical sub-sampling):

Conv. 1: 256 SRAMs for divided 256 banks.
Conv. 2: 16 SRAMs for divided 16 banks with twice of

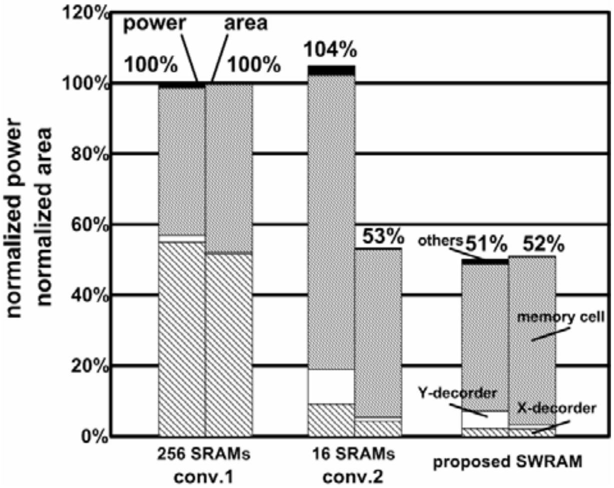


Fig. 33 Area and power comparisons.

Fig. 34 Layout design of a bank of SWRAM.

cycle count and a proper pixel choice.

Figure 33 shows normalized power per access and area. The area estimation includes area overhead by routing congestion. The area and the power of X-decoders are more than 50% with conv. 1. The area of conv. 2 has the high power consumption because of 2 cycle times of accesses. The proposed SWRAM reduced the power by 49% and the area by 48% compared with conv. 1. Figure 34 shows the layout design of SRAM bank.

5. VLSI Implementation

Figure 35 shows a chip layout of the ME processor core in a 90-nm CMOS technology. A 410-Kbits 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 $2.5 \times 2.5\,\text{mm}^2$. The static timing analysis after back annotation shows that 150 MHz operation is attained at a nominal supply voltage of 1.0 V. The chip specification is shown in Table 3.

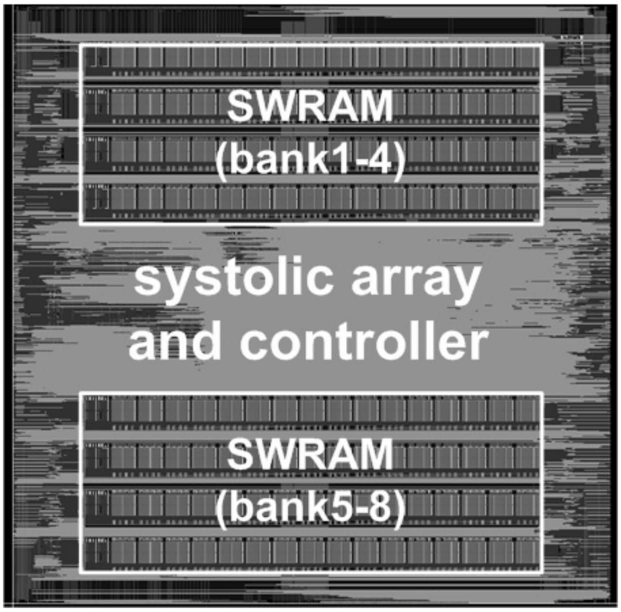


Fig. 35 Chip layout of proposed IME processor core.

Table 3 Chip specification.

Technology	90 nm
Chip size	2.5 x 2.5 mm^2/ref. frame in HDTV
Voltage	1.0 V
Max freq.	150 MHz @ 1.0V
Search range	±128 × ±64 max
Memory size	SWRAM: 410 Kbits
Power	HDTV : 48 mW @ 1 ref. frame 100MHz 1.0V 96 mW @ 2 ref. frames 100MHz 1.0V SDTV : 14 mW @ 1 ref. frames 17MHz 1.0V 20 mW @ 2 ref. frames 34MHz 1.0V 28 mW @ 3 ref. frames 50MHz 1.0V

The gate-level power estimation accompanied with circuit simulation for memory circuit has been performed over the detail. Here a commercial power estimation tool was employed for the whole processor core circuit with physical parameter after place & routing. Also test pattern was generated so as to execute the proposed algorithm. A 100 MHz operation supports one-reference-frame search with a 30-fps HDTV resolution video by one core which dissipates 48 mW at 1 V. Two-core configuration consumes 96 mW for two-reference-frame search with the same resolution video at 100 MHz and 1 V condition. For SDTV resolution, several number of reference frames are supported to control picture quality with only one core. Power dissipations in SDTV resolution are 14 mW at 17 MHz for one reference frame, 20 mW at 34 MHz for two reference frames, 28 mW at 50 MHz for 3 reference frames. The proposed processor realizes real time motion detection for main profile encoding with MBAFF mode on lower power and smaller area in comparison with existing solutions referred in Sect. 1.

6. Conclusion

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

The proposed algorithm is the hierarchical algorithm of the fine search and coarse search. The fine search is adaptively carried out, based on an image analysis result obtained by the coarse search. The workload is reduced by 96%, and the degradation of picture quality does not appear in the proposed algorithm. The reconfigurable ring-connected systolic array architecture provides the minimal amount of transferred data and low computation cycles with the coarse and fine searches in the algorithm. The power and the area of a search window buffer are reduced by the proposed SWRAM which supports the rectangular access, segmentation-free access and sub-sampling access.

The processor has been designed by 90 nm CMOS design-rule. The gate-level power estimation accompanied with circuit simulation demonstrates sub-100 mW realization of H.264 main profile motion estimation processor core for MBAFF encoding.

Acknowledgments

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. Authors thank Mr. Tetsuya Kamino and Mr. Kosuke Mizuno for their design efforts.

References

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

[2] 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, June 2006.

[3] Y.-W. Huang, T.-C. Chien, C.-H. Tsai, C.-Y. Chen, T.-W. Chen, C.-S. Chen, C.-F. Shen, S.-Y. Ma. T.-C. Wang, B.-Y. Hsieh, H.-C. Fang, and L.-G. Chen, "A 1.3TOPS H.264/AVC single chip encoder for HDTV applications," ISSCC Dig. Tech. Papers, pp.128–129, Feb. 2005.

[4] K. Kumagai, C. Yang, H. Izumino, N. Narita, K. Shinjo, S. Iwashita, Y. Nakaoka, T. Kawamura, H. Komabashiri, T. Minato, A. Ambo, T. Suzuki, Z. Liu, Y. Song, S. Goto, T. Ikenaga, Y. Mabuchi, and K. Yoshida, "System-in-silicon architecture and its application to H.264/AVC motion estimation for 1080HDTV," ISSCC Dig. Tech. Papers, pp.430–431, Feb. 2006.

[5] Z. Liu, Y. Song, M. Shao, S. Li, L. Li, S. Ishiwata, M. Nakagawa, S. Goto, and T. Ikenaga, "A 1.41 W H.264/AVC REAL-TIME ENCODER SOC FOR HDTV1080P," Symp. VLSI Circuit Dig. Tech. papers, pp.12–13, June 2007.

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

[7] Y. Murachi, M. Hamano, T. Matsuno, M. Miyakoshi, M. Miyama, and M. Yoshimoto, "A 95 mW 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.

[8] W.I. Choi, B. Jeon, and J. Jeong, "Fast motion estimation with modified diamond search for variable motion block size," IEEE International Conference on Image Processing (ICIP), vol.3, pp.371–374, Sept. 2003.

[9] L. Yang, K. Yu, J. Li, and S. Li, "An effective variable block-size early termination algorithm for H.264 video coding," IEEE Trans. Circuits Syst. Video Technol., vol.15, no.6, pp.784–788, June 2005.

[10] R. Srinivasan and K. Rao, "Predictive coding based on efficient motion estimation," IEEE Trans. Commun., vol.33, no.8, pp.888–896, Aug. 1985.

[11] JM 9.8, http://iphome.hhi.de/suehring/tml/

[12] J. Miyakoshi, Y. Murachi, T. Ishihara, H. Kawaguchi, and M. Yoshimoto, "A power- and area-efficent 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, Nov. 2006.

Yuichiro Murachi was born on November 1, 1980. He received a B.S. degree from Kanazawa University in 2003. He received an M.E. degree from Kanazawa University, Ishikawa, Japan, in 2005. He is currently enrolled in the doctoral course at Kobe University. His research interests are VLSI systems and implementation of multimedia communication systems.

Junichi Miyakoshi received the Ph.D. degree in computer science from Kobe University, Hyogo, Japan, in 2007. His current research interests include high-performance and low-power multimedia VLSI designs. He is a member of IEEE.

Masaki Hamamoto received a M.E. degree from Kobe University, Hyogo, Japan in 2007. His research interests include low-power VLSI algorithms and architectures, and multi-media signal processing.

Takahiro Iinuma received the B.S. in Computer and Systems Engineering from Kobe University, Hyogo, Japan in 2006. He is currently on the master course at Kobe University. Since 2005, he has been involved in the research and development of low-power multimedia VLSI.

Tomokazu Ishihara was born on December 28, 1983. He received a B.E. degree from Kobe University, Hyogo, Japan in 2006. He is currently on the master course at Kobe University. His research interests include low-power VLSI algorithms and architectures, and multimedia signal processing.

Fang Yin received the B.S. from Anhui University of Finance and Economics. She is currently on the master course at Kobe University. Since 2005, she has been involved in the research and development of low-power multimedia VLSI.

Jangchung Lee received the B.S. degree from Kobe University, Hyogo, Japan in 2007. His research interests include low-power VLSI algorithms and architectures, and multi-media signal processing.

Hiroshi Kawaguchi received B.E. and M.E. degrees in Electronic Engineering from Chiba University, Chiba, Japan, respectively, in 1991 and 1993. He received a Ph.D. degree in Engineering from the University of Tokyo, Tokyo, Japan, in 2006. He joined Konami Corp., Kobe, Japan, in 1993, where he developed arcade entertainment systems. He moved to the Institute of Industrial Science, the University of Tokyo, as a Technical Associate in 1996, and was appointed a Research Associate in 2003. In 2005, he moved to the Department of Computer and Systems Engineering, Kobe University, Kobe, Japan, as a Research Associate. Since 2007, he has been an Associate Professor at the Department of Computer Science and Systems Engineering, Kobe University. He is also a Collaborative Researcher with the Institute of Industrial Science, the University of Tokyo. His current research interests include low-power VLSI design, hardware design for wireless sensor networks, and recognition processors. Dr. Kawaguchi received the IEEE ISSCC 2004 Takuo Sugano Outstanding Paper Award and the IEEE Kansai Section 2006 Gold Award. He has served as a Program Committee Member for IEEE Symposium on Low-Power and High-Speed Chips (COOL Chips), and as a Guest Associate Editor of IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences. He is a member of IEEE and ACM.

Masahiko Yoshimoto received a B.S. degree in Electronic Engineering from the Nagoya Institute of Technology, Nagoya, Japan, in 1975, and an M.S. degree in Electronic Engineering from Nagoya University, Nagoya, Japan, in 1977. He received a Ph.D. degree in Electrical Engineering from Nagoya University, Nagoya, Japan in 1998. He joined the LSI Laboratory, Mitsubishi Electric Power Products Inc., Itami, Japan, in April 1977. During 1978–1983 he was engaged in the design of NMOS and CMOS static RAM, including a 64 K full CMOS RAM with the world's first divided-word-line structure. From 1984, he was involved in research and development of multimedia ULSI systems for digital broadcasting and digital communication systems based on MPEG2 and MPEG4 Codec LSI core technology. Since 2000, he has been a Professor of the Dept. of Electrical and Electronic Systems Engineering at Kanazawa University, Japan. Since 2004, he has been a Professor of the Dept. of Computer and Systems Engineering at Kobe University, Japan. His current activities specifically emphasize research and development of multimedia and ubiquitous media VLSI systems including an ultra-low-power image compression processor and a low-power wireless interface circuit. He holds 70 registered patents. He served on the Program Committee of the IEEE International Solid State Circuit Conference during 1991–1993. In addition, he has served as a Guest Editor for special issues on Low-Power System LSI, IP, and Related Technologies of IEICE Transactions in 2004. He received R&D100 awards from R&D Magazine in 1990 and 1996, respectively, for development of the DISP and development of a real-time MPEG2 video encoder chipset.