Fractional Motion Estimation Hardware Circuits for H.264/ Advanced Video Coding

Chien-Min Ou and Chiung-Chou Liao

Abstract—**A novel half-pel full-search motion estimation VLSI architecture for H.264/AVC video encoders is presented in this paper. Based on the processing element arrays eliminating redundant data accesses and attaining 100 % utilization, the architecture can be implemented with low clock rate while having high processing throughput. Such an implementation is particularly suited to applications requiring real time operations with high compression efficiency and low power.**

Index Terms—**VLSI hardware architecture, Video coding, Fractional motion estimation, H.264 standard.**

I. Introduction

Motion estimation (ME) [1]-[3] and compensation based on block-matching operations have been extensively used for removing temporal redundancy in many video coding applications because of their simplicity and effectiveness. Although the integer pel ME may provide satisfactory reconstruction results, the fractional pel ME is necessary in many applications because of the increasing demand on high video compression quality. One conventional approach to realizing the fractional ME in hardware is a direct extension to the those for the VLSI [4] implementation of integer pel ME, where the same processing element (PE) array for the integer pel ME is also used for the fractional pel ME [5]. In this approach, the interpolated samples in the search region will first be computed, and stored in a memory buffer for subsequent accesses. This may introduce large storage overhead. In the PE array, the search for the candidate block having minimum sum of absolute difference (SAD) should cover the candidate blocks formed by integer position and interpolated samples. Accordingly, as compared with the integer pel ME, the fractional pel ME based on this architecture has longer latency for identifying the optimal candidate block. Although the long latency can be compensated by increasing the clock rate and the source voltage of the circuit, the average power may also be increased.

The objective of this paper is to present a novel VLSI architecture for half-pel ME of H.264/AVC [1], [6]-[10] having the advantages of low storage overhead, low latency and low power. In the architecture, low latency and low power are attained by the employment of concurrent PE arrays and the reduction of source voltage

and clock rate. That is, we lower the power consumption and compensate the delay by increasing the silicon area [11]. There are four PE arrays in the architecture. These arrays are responsible respectively for the SAD computation of candidate blocks formed by integer position samples, vertically interpolated half-pel samples, horizontally interpolated half-pel samples, and diagonally interpolated half-pel samples. Each PE array is able to eliminate redundant accesses among adjacent candidate blocks so that it is not necessary to store the interpolated samples in the memory buffer. The storage overhead for fractional ME therefore can be minimized. The proposed architecture has been prototyped, simulated and synthesized for 0.18 μm CMOS technology using UMC standard cells. The measured data demonstrate that the architecture can be an effective alternative for the applications where high video compression quality, high computational speed and low power are desired.

II. Background

This section reviews some background material of this paper. We start with the integer-pel ME. Fig. 1 shows a $N \times N$ current block ($N=4$) and its search area for the integer pel full-search BMA. The range of displacement is $[-p, -(p-1)]$ ($p=2$) in both $x-$ and $y-$ directions. Therefore, the size of the search region is given by $(N+2p-1) \times (N+2p-1)$ There are $2p \times 2p$ candidate blocks in the search area. The candidate blocks in the same row form a block strip. Adjacent block strips are overlapping. For the illustration purpose, the columns of the block strips are indexed as shown in Fig. 1.

The 1D systolic array [12] of the full-search BMA is shown in Fig. 2, which skews each column of the current blocks and candidate blocks for the SAD computation. Table I shows data flow schedule indicating the starting clock for calculating each column SAD, which will take N clock cycles to complete. Every N consecutive column SADs will then be accumulated as one block SAD.

III. The Proposed Architecture

Although the 1D systolic array is simple to construct, the columns of search area will be re-fetched as shown in Fig. 2 and Table I. Therefore, it is necessary to use memory buffers for data reuse.

The proposed architecture is based on the PE array presented in our previous work [13], which is used for the integer pel full-search ME for an $N \times N$ current block without the redundant accesses of candidate blocks within the same block strip. The circuit contains N 1D systolic arrays, and each systolic array contains N PEs, as shown in Fig. 3.

Clock Cycle	Inputs		Operations
	Current Block	Search Area	
0	X0	Col0	$|X0 - Col0|$
1	X1	Col1	$|X1 - Col1|$
2	X2	Col2	$|X2 - Col2|$
3	X3	Col3	$|X3 - Col3|$
4	X0	Col1	$|X0 - Col1|$
5	X1	Col2	$|X1 - Col2|$
6	X2	Col3	$|X2 - Col3|$
7	X3	Col4	$|X3 - Col4|$

The circuit operates by scheduling the columns of the current block through a delay line, and broadcasting columns of two adjacent candidate block strips in the search region on each clock cycle. Each 1D systolic array then skews the pixels in the input columns for SAD computations.

Table II shows the data flow of the PE array for the current block and its search area with *N*=4 In addition to having high throughput [13], the major advantage of the circuit is that each column in the same block strip is accessed only once. The redundant accesses within each block strip can then be removed. This advantage is very helpful for the half-pel ME.
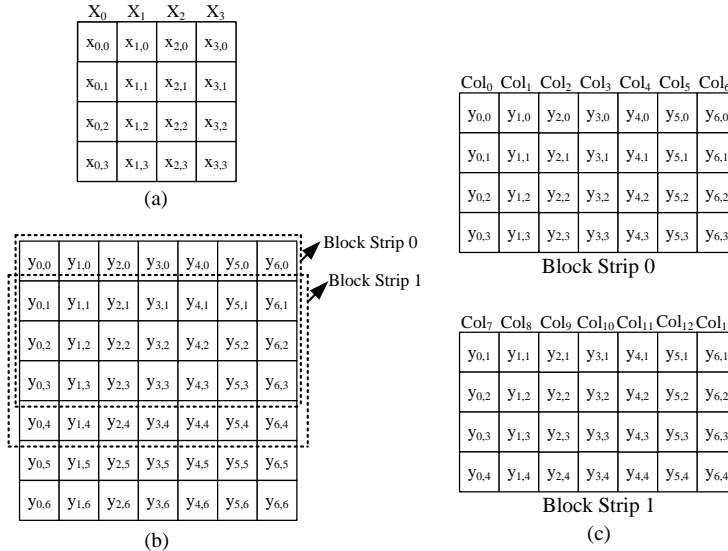

(a) the 4×4 current block, (b) the search area, (c) the index of columns of the first two block strips.
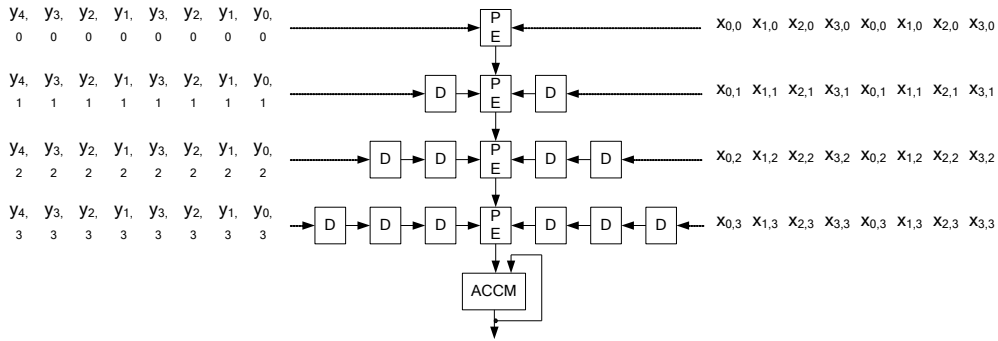Fig. 1. The 4×4 current block and its search area.

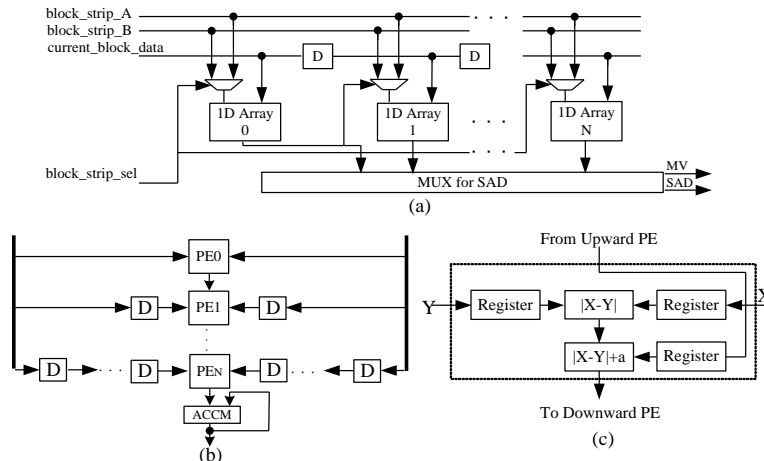Fig. 2. The basic structure of 1D array.

Fig. 3. The VLSI architecture of the PE array for integer pel full-search ME without redundant accesses of candidate blocks within the same block strip: (a) The architecture (b) 1D array architecture (c) PE architecture.

TABLE II: THE DATA FLOW OF THE PROPOSED PE ARRAY SHOWN IN FIGURE 3 FOR $P=2$

Clock	Current_block_data	Block_strip_A	Block_strip_B	1-D Array 0	1-D Array 1	1-D Array 2	1-D Array 3
0	X0	Col0		|X0 − Col0|			
1	X1	Col1		|X1 − Col1|	|X0 − Col1|		
2	X2	Col2		|X2− Col2|	|X1 − Col2|	|X0 − Col2|	
3	X3	Col3		|X3 − Col3|	|X2 − Col3|	|X1 − Col3|	|X0 − Col3|
4		Col4	Col7	|X0 − Col7|	|X3 − Col4|	|X2 − Col4|	|X1 − Col4|
5		Col5	Col8	|X1 − Col8|	|X0 − Col8|	|X3 − Col5|	|X2 − Col5|
6		Col6	Col9	|X2 − Col9|	|X1 − Col9|	|X0 − Col9|	|X3 − Col6|
7			Col10	|X3 − Col10|	|X2 − Col10|	|X1 − Col10|	|X0 − Col10|
8		Col14	Col11	|X0 − Col14|	|X3 − Col11|	|X2 − Col11|	|X1 − Col11|
9		Col15	Col12	|X1 − Col15|	|X0 − Col15|	|X3 − Col12|	|X2 − Col12|
10		Col16	Col13	|X2 − Col16|	|X1 − Col16|	|X0 − Col16|	|X3 − Col13|
11		Col17		|X3 − Col17|	|X2 − Col17|	|X1 − Col17|	|X0 − Col17|
…	…	…	…	…	…	…	…

In addition, from Table II, it can be observed that the PE array produces one block SAD for each clock cycle. Define the latency of the structure as the total number of clock cycles required for identifying the candidate block having minimum SAD for each current block. Accordingly, the latency of the PE array is $2p \times 2p$ The latency of conventional 1D array shown in Fig. 2 is $N \times 2p \times (2p+N-1)$ [12]. The proposed architecture therefore has lower latency over the basic 1D array.

In the H.264 half-pel ME, each half-pel sample that is horizontally or vertically adjacent to two integer samples is interpolated from integer-position samples using 6-tap finite impulse response (FIR) filter [11]. The remaining half-pel samples (termed diagonal half-pel samples) are then calculated by interpolating between six horizontal or vertical half-pel samples. Fig. 4 shows the proposed architecture for the realizing the H.264 half-pel ME. It contains the FIR filter bank, and the PE arrays. The FIR filter bank calculates the horizontal, vertical and diagonal half-pel samples from the integer-position samples. The four PE arrays are then responsible for the subsequent SAD computation. All the PE arrays have identical architecture shown in Fig. 3.
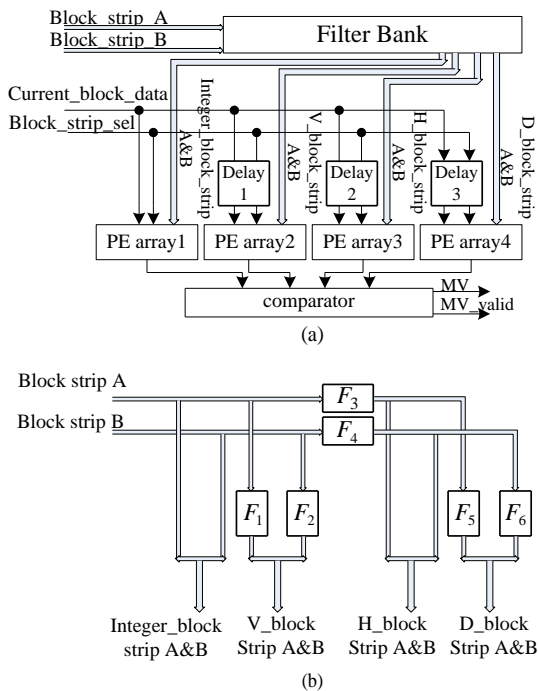
To compute the horizontal, vertical and diagonal half-pel samples, two adjacent candidate block strips of integer-position samples (denoted by Block_strip_A and Block_strip_B in Fig. 4) are accessed in the manner similar to the schedule shown in Table II. Each source block strip is interpolated to form block strips of horizontal, vertical and diagonal half-pel samples by the filter bank. Since two adjacent source block strips are accessed concurrently, and each one is used to calculate three interpolated strips, the filter bank produces six interpolated block strips. It consists of six filters: F_j, $j=1, …, 6$ (depicted in Fig. 4.(b)). Each filter produces one column of the interpolated samples at a time for the subsequent PE array. The filters F_1 and F_2 operate on each single column of the input strips for generating the vertically interpolated strips on every clock cycle. The filters F_3 and F_4 perform the averaging operations across columns of the input strips for calculating the horizontally interpolated strips. Finally, the filters F_5 and F_6 are used to produce the diagonally interpolated strips by averaging pels on each single column of horizontally interpolated strips.

All the filter outputs are delivered directly to the PE arrays. It is not necessary to store the interpolated pels, because the PE arrays require no redundant access within a block strip. This can effectively reduce the storage overhead and power consumption. In addition, since four PE arrays operate concurrently, the SAD computation of integer-position samples can be performed in parallel with those of horizontal, vertical and diagonal half-pel samples.

TABLE III: THE PROPOSED FRACTIONAL ME CHIP PERFORMANCE

Number of PE	$4 \times 8 \times 8$
Range of displacement	[8,-7]
Block size	8×8
Technology	UMC 0.18 μm
Gate count	351K
Max frequency	334 MHZ

TABLE IV: COMPARISONS OF CLOCK PATES AND AVERAGE POWER CONSUMPTION

Frame Size		CIF	4CIF	16CIF	SDTV	HDTV
Frame Rate (fps)		30	30	30	30	60
Proposed Circuit	Clock Rate (MHz)	12.16	48.64	194.64	110.59	221.18
	Power (mW)	33.23	135.95	453.15	271.91	679.51
Single-PE	Clock Rate (MHz)	48.64	194.64	778.56	442.36	884.72
	Power (mW)	66.58	251.93	887.72	531.55	1329.20

(a) The architecture, (b) The architecture of filter bank.
Fig. 4. Proposed half-pel ME architecture.

Accordingly, the latency of the proposed architecture is $2p \times 2p$ which is identical to that of the architecture in Fig. 3 for integer-pel ME. The architecture therefore attains low latency for fractional ME with low power consumption and low storage size overhead.

IV. THE EXPERIMENTAL RESULTS

The H.264 ME chip based on the proposed architecture was designed with Synopsys synthesis tools using a standard cell library based on the UMC 0.18 μm CMOS technology process. The main characteristics of the circuit are presented in Table III. The circuit contains $4 \times 8 \times 8$ PEs (N=8), with a range of displacement [8,−7] (p=8). The latency and maximum frequency of the circuit are given by 256 (i.e., $4p^2$) and 334 MHz, respectively.

Table IV shows the required clock rates and the corresponding average power dissipation of the proposed architecture for various frame sizes and frame rates. It can be observed from the table that the clock rate for processing the high definition TV (HDTV) video sequences at 60 frames/sec is only 238 MHz, which is still lower than the maximum clock rate of the circuit as shown in Table III. This chip therefore supports a wide range of video formats for H.264 based applications.

One major advantage of high throughput is that the clock rate is substantially reduced subject to a constraint on frame size and frame rate. The clock rate reduction may lower the power consumption [1] for ME, and is useful for low power applications.

To justify the employment of 4 PE arrays in our architecture, the clock rates and the power

dissipation of the architecture containing only one PE array (termed single PE system) for half-pel ME are also included in the Table IV. In the single PE system, the integer block strip, and horizontally, vertically and diagonally interpolated strips are processed one at a time. In addition, the system should contain local RAM for storing the interpolated samples. Consequently, as shown Table IV, our architecture has lower power dissipation than the single PE system subject to the same video format. For example, the power consumption of our architecture is only 679.51 mW for HDTV sequences. On the contrary, the power consumption of single PE system is 884.72 mW for the same sequence. All these facts demonstrate the effectiveness of the proposed architecture.

V. CONCLUSION

As compared with the single PE architecture for fractional ME, the proposed architecture is able to produce the best MVs with lower clock rate. Therefore, the circuit is well suited for low power designs. In particular, the clock rate for the VBS-BMA operations over CIF sequences at 30 fps is 12.16 MHz. The resulting power dissipation is only 33.23 mW, which may be attractive for mobile or portable video applications. On the other hand, the frame size and frame rate supported by the circuit can also be substantially extended subject to a clock rate constraint. In our experiment, the required clock rate for HDTV sequences at 60 fps is 221.18 MHz. The circuit may therefore be very helpful for designs requiring high

visual quality. Gate-level synthesization and verification illustrate that our circuit is beneficial for enhancing the performance of H.264 encoders over a wide range of video applications.

REFERENCES

[1] I. E. G. Richardson, *H.264 and MPEG-4 Video Compression*, John Wiley & Sons, 2003.

[2] T. Ogunfunmi, O. Ndili, and P. Arnaudov, "On low power fractional motion estimation algorithms for H.264," 2012 IEEE Workshop on Signal Processing Systems (SiPS), 2012, pp. 103–108.

[3] G. L. Li and T. S. Chang, "An efficient mode pre-selection algorithm for H.264/AVC scalable video extension fractional motion estimation," in *Proc. 2011 17th International Conference on Digital Signal Processing (DSP)*, 2011, pp. 1-5.

[4] W. J. Hwang, C. M. Ou, W. M. Lu, and C. F. Lin, "VLSI architecture for motion vector quantization," *IEEE Transactions on Consumer Electronics*, vol. 49, no. 1, Feb. 2003, pp. 237-242.

[5] M. Sayed and W. Badawy, "A half-pel motion estimation architecture for MPEG-4 applications," in *Proc. IEEE Int. Symp. on Circuits and Systems*, vol. 2, 2003, pp. 792-795.

[6] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra. "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 13, 2003, pp. 560-576.

[7] M. Guarisco, H. Rabah, and A. Amira, "Dynamically reconfigurable architecture for real time adaptation of H264/AVC-SVC video streams," Computer Vision and Pattern Recognition Workshops (CVPRW), 2010, pp. 39-44.

[8] M. H. Karamiata, A. Shariati, H. Taheri, and F. S. Khodadad, "An advanced surveillance-advertiser data network for E-city management using "H264/AVC-Pseudo-PWM", in *Proc. Technique, Developments in E-systems Engineering (DeSE)*, 2011, pp. 387-391.

[9] F. Wang, Y. Li, H. Jia, X. Xie, and W. Gao, "An efficient fractional motion estimation architecture for avs real-time full hd video encoder," in *Proc. 2012 IEEE International Conference on Imaging Systems and Techniques (IST)*, 2012, pp. 279 - 284.

[10] M. Sun, L. Li, J. Wang, H. Pan, and W. Li, "A horizontal data reuse approach for fractional motion estimation in H.264/AVC encoder," in *Proc. 2012 International Conference on Computer Science and Information Processing (CSIP)*, 2012, pp. 821-825

[11] A. P. Chandrakasan and R. W. Brodersen, "Minimizing power consumption in digital CMOS circuits," in *Proc. The IEEE*, vol. 83, 1995, pp. 498-523.

[12] P. Kuhn, "Algorithms, complexity analysis and VLSI architectures for MPEG-4 motion estimation," *Kluwer Academic*, 1999

[13] C. M. Ou and W. J. Hwang, "An efficient VLSI architecture for H.264 variable block size motion estimation," in *Proc. IEEE Trans. Consumer Electronics*, vol. 51, 2005, pp. 1291–1299.

Chien-Min Ou received his diploma in Telecommunication Engineering from Chien Shin Institute of Technology, Chung Li, Taiwan, R.O.C. in 1978 and M.S., and Ph.D. degrees in Electrical Engineering from Chung Yuan Christian University, Chung Li, Taiwan, in 2000, and 2003, respectively. From 1978 to 1995, he worked as a Teach Assistant in the Department of Electronics Engineering, Chien Hsin Institute of Technology, Chung Li, Taiwan, R.O.C. In 1995, he was a Lecturer with the Department of Electronics Engineering, Ching Yun Institute of Technology, Chung Li, Taiwan, R.O.C. From 2004 to 2010, he was an Assistant Professor of the Department of Electronics Engineering, Ching Yun University, Chung Li, Taiwan, R.O.C. From 2010 to 2011, he became a Associate Professor of the same department. He is currently a Associate Professor of the Department of Electronics Engineering, at the Chien Hsin University of Science and Technology. He is also a member of the honor society Phi Tau Phi. He research topics include Genetic Algorithm, Artificial neural networks, Pattern recognition, Hardware implementation, VLSI design and testing, Image processing, Video compression, Motion estimation.

Chiung-Chou Liao was born in Yung-Lin, Taiwan, on October 6, 1973. He received the B.S., M.S., and Ph.D. degrees from Chung Yuan Christian University, Zhongli, Taiwan, in 1996, 1998, and 2001, respectively. Since 2010, he has been an Associate Professor with the Department of Electronic Engineering, Chien Hsin University, Zhongli. He is a reviewer of the IET Generation, Transmission and Distribution. His current research interests include renewable energy and power quality of power systems.