

An advanced Parallel FPGA Architecture for Bi-directional Motion Estimation*

Yangfan Huang, Minjun Deng, Donglian Li, Zhenzhen Li, Mingyan Yu, Cailan Zeng, Yu Zhang, Zhuo Chen, Pengcheng Cao and Ran Liu

College of Communication Engineering, Chongqing University, Chongqing, 400030, China

hyf@cqu.edu.cn, lizhenzhen_1116@163.com, cao2008cheng@163.com, ran.liu_cqu@qq.com

Abstract

Motion estimation (ME) and motion compensation (MC) are the key elements for frame rate up-conversion (FRUC) system. Fast and accurate motion estimation is the premise of high quality motion compensation. Unlike conventional unidirectional motion estimation which brings holes, overlaps and blocking artifacts, the bi-directional motion estimation does not produce any overlapped pixel or hole in the interpolated frames. As a result, the bi-directional motion estimation has better performance than conventional unidirectional motion estimation. This paper presents an efficient FPGA architecture targeting bi-directional motion estimation hardware implementation. This proposed architecture can achieve real-time processing for 1280x720@60Hz under 200MHz operating frequency. The design is described in Verilog HDL, verified in Virtex5 FPGA platform. Experimental results show that the proposed architecture has high performance and low cost for bi-directional motion estimation algorithm. This architecture can be used for video post-processing system.

Keywords: *Frame Rate Up-Conversion, ME/MC, Bi-Directional Motion Estimation, FPGA*

1. Introduction

With the development of digital television, especially large-scale display device, higher and higher requirements are put forward on video image processing technology. When displaying dynamic pictures on LCD, the problem such as tailing, jitter and blurring usually occurs. To solve these problems, frame rate up conversion (FRUC) technology arises spontaneously. FRUC can achieve frame rate conversion between different video scanning formats, so as to reduce the holding time of LCD and eliminate the above problems effectively [1]. For example, the frame rate can be increased from 50/60fps to 100/120fps to make the display more fluent. Thus, FRUC has become a key technique in the development of television technology.

Motion estimation and motion compensation are the key algorithms for video compression encoding and video post-processing [2]. ME plays an important role in the ME/MC FRUC system; the accuracy of ME could directly affect the quality of interpolated frames. Fast and accurate motion estimation is the premise for high quality motion compensation [3]. However, with video solution increased, such as HD 720p or 1080p, the computational complexity of motion estimation algorithm will dramatically increase. Generally, in hardware implementation, motion estimation normally spends about 70% resource in video processing system [4]. How to improve the speed and precision of motion estimator, and how to simplify the corresponding hardware structure has always been a hot research area in the FRUC system, or even in the video encoding and video processing aspects.

Many VLSI architectures have been proposed to implement the motion estimator based on full-search block-matching arithmetic (FS_BMA) algorithm. To meet the requirement of parallel and high-speed processing, processing element (PE) array are adopted in most of the architectures. Ref [5] proposed a FS_BMA motion estimator based on frame-level pipelining. Ref [6] proposed a cascaded motion estimator structure with low resource consumption and low bandwidth requirements. Ref [7] proposed a VLSI architecture for unidirectional motion estimation, which is based on the structure of parity array registers. The architecture can reduce the access times of external memory effectively by data multiplex. Many of the above architectures are based on unidirectional motion estimation. However, when applied to the FRUC system, unidirectional motion estimation will result in holes and overlaps [8]. Bi-directional motion estimation can effectively cope with these problems. In the VLSI architecture of this paper, efficient data multiplex is adopted to use the duplicate data. Pipelining technology is adopted to improve the operating frequency. The final result reveal that this architecture can support the real time processing of 720p video under the 200 MHz clock frequency with low resource consumption.

The remaining portion of this paper is organized as follows. First, the theory of bi-directional motion estimation is briefly introduced in Section 2. Then, Section 3 presents the VLSI architecture. With the proposed architecture, Section 4 demonstrates the implementation results and corresponding simulation. Finally, Section 5 concludes the paper.

2. Principle of Bi-directional Motion Estimation

2.1. Bi-directional Motion Estimation

Most of the current motion estimators are based on unidirectional motion estimation, but unidirectional motion estimation will bring holes and overlaps when applied to the FRUC system. The procedure of the FRUC system based on unidirectional motion estimation can be separated into three steps [8]:

- Step1: divide the previous frame (FP) into several non-overlapping blocks;
- Step2: find the most matching block for each block of FP from the next frame (FN);
- Step3: determine the position of the inserted block in interpolated frame (FI) corresponding to the matching blocks in FP and FN.

As shown in Figure 1, when searching matching blocks for two adjacent blocks of FP, the matching blocks in FN are overlapped, which will result in the problems that he finally obtained positions of the inserted block in FI are also overlapped. Then, overlap occurs. Similarly, hole occurs when some areas of FI are not covered by any matching blocks. Holes and overlaps are inherent defect of unidirectional motion estimation and hard to be eliminated.

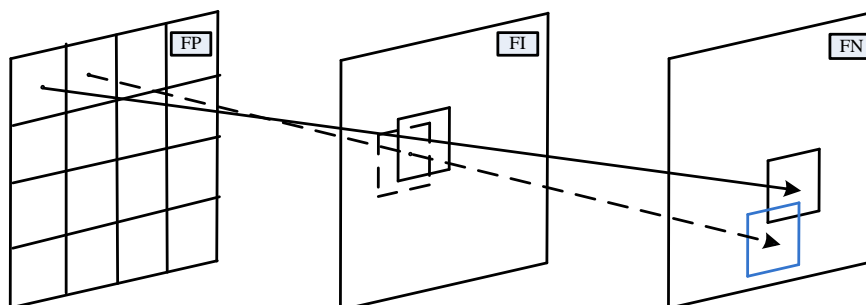


Figure 1. Unidirectional Motion Estimation

Bi-directional motion estimation can effectively cope with the above problems. The procedure of bi-directional motion estimation can be separated into two steps [2]:

Step1: divide FI into several non-overlapping blocks;

Step2: set the blocks of FI as mirror center and find the best match blocks in FP and FN.

As shown in Figure 2, each block of the FI has only one pair of matching block in the FP and FN, which will effectively avoid holes and overlaps.

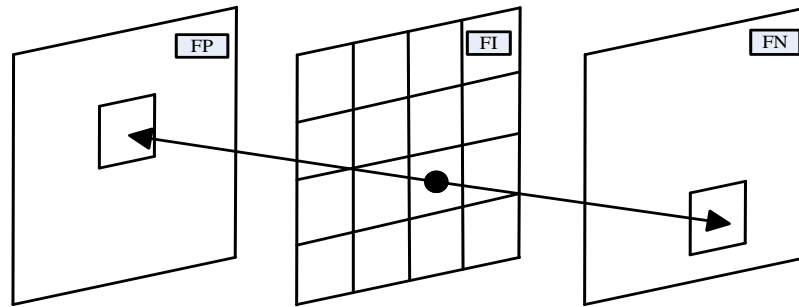


Figure 2. Bi-Directional Motion Estimation

2.2. Local Full Search Algorithm

There are many search algorithms for bi-directional motion estimation, such as full search, three-step, four-step and diamond search algorithm. Three-step [9], four-step [10] and diamond search algorithm [11, 12] are all belong to fast search algorithm. Though the computations of the three algorithms are small, they cannot get the best matching results. On the contrary, full search (FS) algorithm [6] can get the best search results, but the computation is enormous. Based on this, local full search (LFS) algorithm is adopted. Compared with FS algorithm, the size of searching window of LFS is determined by developers themselves. In the actual videos, the ranges of motion of moving objects are moderate and the best matching block can obtained from an appropriate searching window, not a whole image instead. Thus, local full search algorithm is adopted in this paper.

Because of the symmetrical search of bi-directional motion estimation, if the searching range is too large, some textures of the image may be replaced by background. According to the results discussed in Ref [4], the size of block is 16×16 and the searching range is -8 to $+7$. Thus, the searching window is 32×32 , as shown in Figure 3.

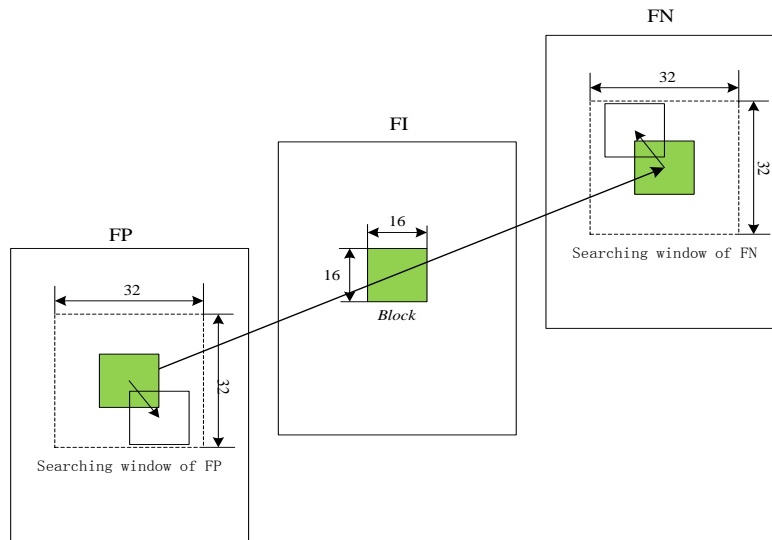


Figure 3. Search Range Of Bi-Directional Motion Estimation

2.3. Matching Criterion

The most common matching criteria for bi-directional motion estimation are MSE (mean square error) criterion and SAD (sum of absolute difference) criterion [2]. MSE criterion is considered to be the best matching criterion because it can express the Euclidean distance between the two blocks. The formulation of MSE criterion is

$$MSE(d_x, d_y) = \frac{1}{M \times N} \sum_{i=x}^{x+M-1} \sum_{j=y}^{y+N-1} [f(i, j, t) - f(i + d_x, j + d_y, t - 1)]^2 \quad (1)$$

Where $M \times N$, represents the size of matching blocks, (d_x, d_y) represents motion vectors, $f(i, j, t)$ represents luminance value at point (i, j) in current image and $f(i + d_x, j + d_y, t - 1)$ represents luminance value at point $(i + d_x, j + d_y)$ in the reference image.

The computational complexity of MSE criteria is quite high, so we use SAD criteria instead.

$$SAD(d_x, d_y) = \sum_{i=x}^{x+M-1} \sum_{j=y}^{y+N-1} |f(i, j, t) - f(i + d_x, j + d_y, t - 1)| \quad (2)$$

3. Improved FPGA Architecture

3.1. Data Multiplex

As described in section 2.2, the size of block is 16×16 , the searching range is -8 to $+7$ and the size of searching window is 32×32 . Here we use a simple example to explain the basic principle of data multiplex. In this instance, the size of block is 4×4 , the searching range is -2 to $+1$ and the size of searching window is 16×16 . Figure 4 shows the relationship between two searching windows of two successive blocks.



Figure 4. Searching Window Of Two Successive Blocks

It can be seen from Figure 4 that half of the two successive searching windows overlap with each other. For such a design with large amount of data, read and write operation of the memory will cost most of the processing time. A method to solve this problem is to reduce the access times of the memory through data multiplex. In this design, the following two aspects are considered: (a) data multiplex on different searching direction of each block; (b) data multiplex between the successive blocks.

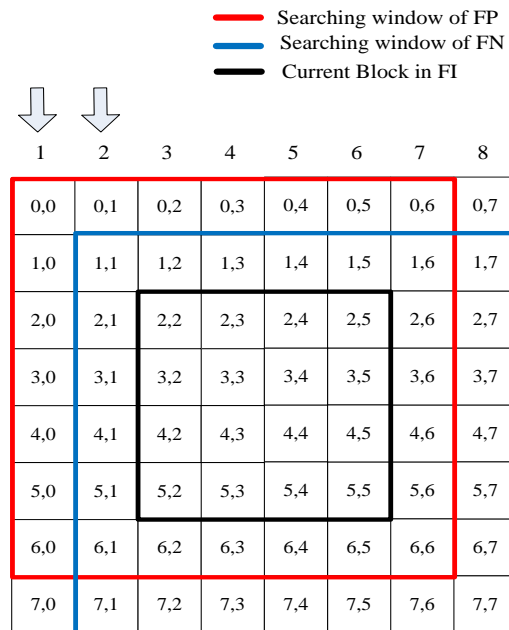


Figure 5. Searching Windows Of The Same Block In FP And FN

Figure 5 depicts the relationships between searching windows in FP and FN. For convenience, the following specifications are stated first:

- (1) The depiction of a block starts with character ‘B’ and the depiction of a pixel starts with character ‘P’;

(2) Pixel coordinate at top left corner of the block is used to depict the block. For example, block B(2,2) means the 4×4 block, which starts from pixel P(2,2) to pixel P(5,5);

(3) FP_, FI_ and FN_ are used as a prefix to distinguish which frame the data belongs to. For example, FN_B (3,3) depicts the 4×4 block in the next frame FN.

In Figure 5, the searching window of block FI_B(2,2) in FP is encircled by red lines and the searching window in FN is encircled by blue lines. According to the search principle of bi-directional motion estimation, blocks FP_B(0,0) ~ FP_B(3,3) in previous FP and blocks FN_B(1,1) ~ FN_B(4,4) are used to calculate SAD values to determine the best matching block for FI_B(2,2). Table 1 shows the relationships between blocks in FP and blocks in FN. As shown in Table 1, there is a one-to-one correlation between blocks in FP and blocks in FN, such as FP_B(0,0) and FN_B(4,4). Then, the SAD values of the sixteen motion vectors are all calculated. The motion vector with the minimum SAD value is the motion vector.

Table 1. Relationships Between Blocks In FP And Blocks In FN

Blocks in FP	Blocks in FN	Motion vector(from FP to FN)
FP_B(0 , 0)	FN_B(4 , 4)	(2, -2)
FP_B(0 , 1)	FN_B(4 , 3)	(1, -2)
FP_B(0 , 2)	FN_B(4 , 2)	(0, -2)
FP_B(0 , 3)	FN_B(4 , 1)	(-1, -2)
FP_B(1 , 0)	FN_B(3 , 4)	(2, -1)
FP_B(1 , 1)	FN_B(3 , 3)	(1, -1)
FP_B(1 , 2)	FN_B(3 , 2)	(0, -1)
FP_B(1 , 3)	FN_B(3 , 1)	(-1, -1)
FP_B(2 , 0)	FN_B(2 , 4)	(2, 0)
FP_B(2 , 1)	FN_B(2 , 3)	(1, 0)
FP_B(2 , 2)	FN_B(2 , 2)	(0, 0)
FP_B(2 , 3)	FN_B(2 , 1)	(-1, 0)
FP_B(3 , 0)	FN_B(1 , 4)	(2, 1)
FP_B(3 , 1)	FN_B(1 , 3)	(1, 1)
FP_B(3 , 2)	FN_B(1 , 2)	(0, 1)
FP_B(3 , 3)	FN_B(1 , 1)	(-1, 1)

3.2. FPGA Architecture

Based on the description above, the corresponding architecture is depicted in Figure 6. The motion estimator is mainly composed of block ram memory unit and calculating array; the block ram memory unit is mainly responsible for getting the image and divides the pixels of FP and FN into four columns according to the odd-even columns. Figure 7 shows the FPGA architecture for calculating array, in this architecture, there are sixteen processing element (PE) arrays named pe0 to pe15 to calculate SAD values. Each PE is responsible for calculating the SAD value of one motion vector. After that, the “compare” component will compare the 16 SAD values and find out the minimum one.

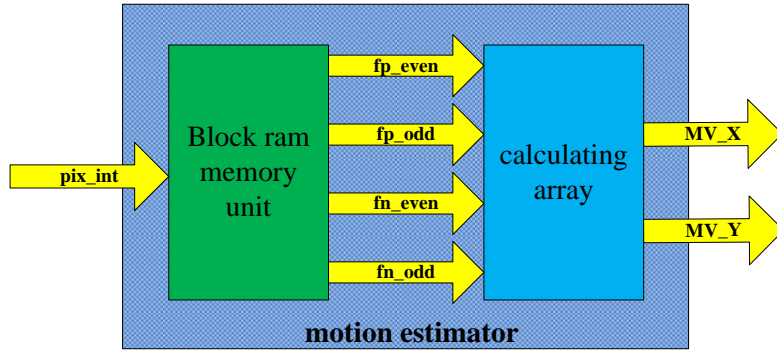


Figure 6. Architecture Of Motion Estimator

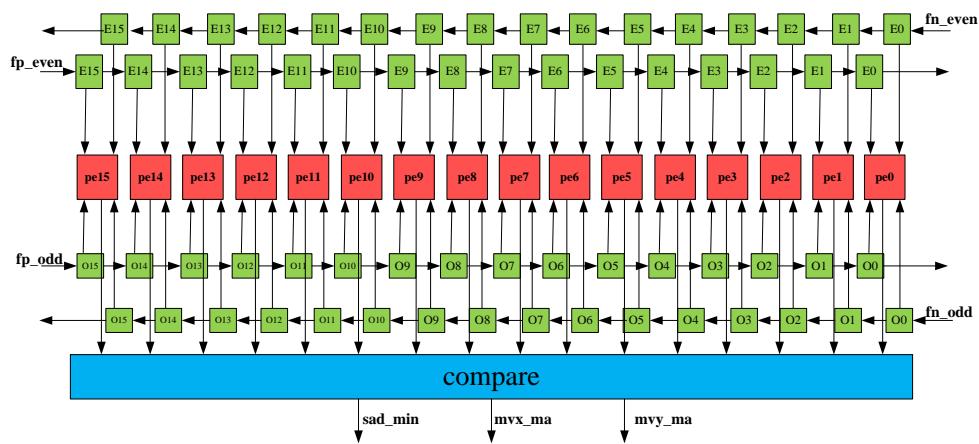


Figure 7. FPGA Architecture for Calculating Array

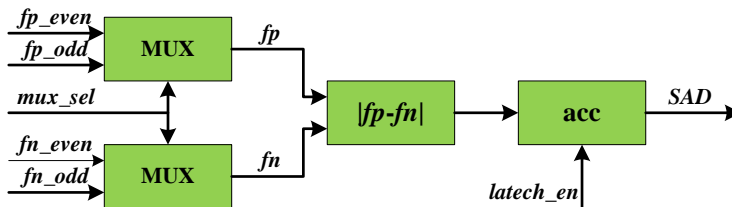


Figure 8. Internal Structure Of PE

Figure 8 shows the internal structure of PE units. Firstly, PE selects the data of odd row or even row according to the signal *mux_sel*. Then, the data of FP and the data of FN are used to calculate the SAD value according to SAD matching criterion.

To fully utilize the overlapping data of the searching window, the motion estimator puts forward a parity column scanning strategy for data distribution. The data is divided into four groups according to the parity columns of FP and FN. Each group is flowed into a shift register group. When motion estimator is operating on the next macro block, the first few columns of data are still stored in the register group. With this strategy, the motion estimator can reuse the overlapping data and reduce the access times of memory.

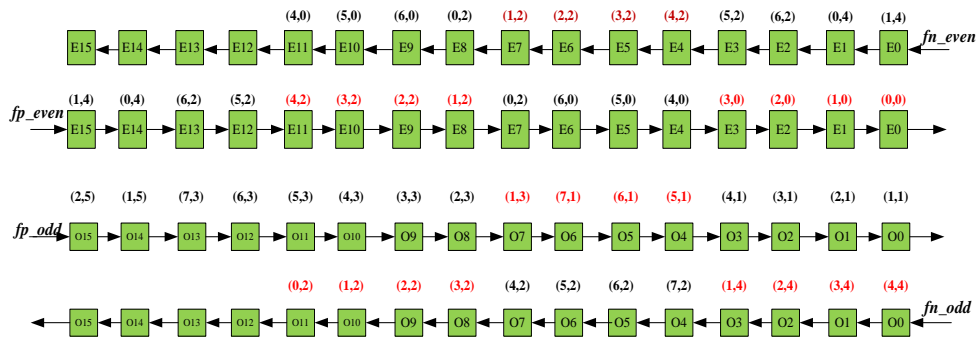


Figure 9. Data Distribution Order Of Register Group

Figure 9 shows the data distribution order of odd array register groups and even array register groups. When calculations for the macro block FI_B(2,2) are finished, the operations for block FI_B(2,6) will start. At this time, the needed data FP_P(0,4), FP_P(1,4), FP_P(6,4), FP_P(0,5), ... FP_P(6,6) (the overlapping data) for block FI_B(2,2) are still stored in the register groups, which are also the needed data for the next block FI_B(2,6). Thus, this architecture can effectively meet the requirement of data multiplex and improve the data utilization.

4. Synthesis Results and Experimental Results

4.1. Synthesis Results

The proposed architecture has been have been described in Verilog HDL and synthesized to Xilinx FPGAs. According to the description, a great deal of logical resources will be needed. The chosen FPGA chip has abundant logical resources for this architecture. Besides, the chip has two HDMI input ports and one HDMI output port, which can support HD or even FULL HD resolution at 60Hz frame rate.

The Xilinx software ISE was used to implement the design. After steps of synthesis, translation, map and place & route, we can get the accumulative hardware resource used in the system. Table 2 presents a summary of the synthesis results for the proposed architecture, considering Xilinx xc5v1x330t FPGA.

Table 2. Device Utilization Summary of the Proposed Architecture

Logic Sources Type	Used	Total	Utilization
Number of Slice Registers	31969	207360	15%
Number of Slice LUTs	34418	207360	16%
Number of fully used LUT-FF pairs	21519	44868	47%
Number of boned IOBs	82	960	8%
Number of Block RAM/FIFO	6	324	1%
Number of BUFG/BUFGCTRLS	2	32	6%

4.2. Experimental Results

The system of bi-directional motion estimation is realized on a Xilinx Virtex-5 xc5v1x330t FPGA. In the experiments, the system captures the image data from DVD player, up the frame rate of the image data real time, and then displayed the processed image on the LCD. The results of bi-directional motion estimation system are shown in Figure 10, the middle pictures are interpolated frames, and the left and right pictures are

original frames. Corresponding, the performance of the implementation is depicted in Table 3.

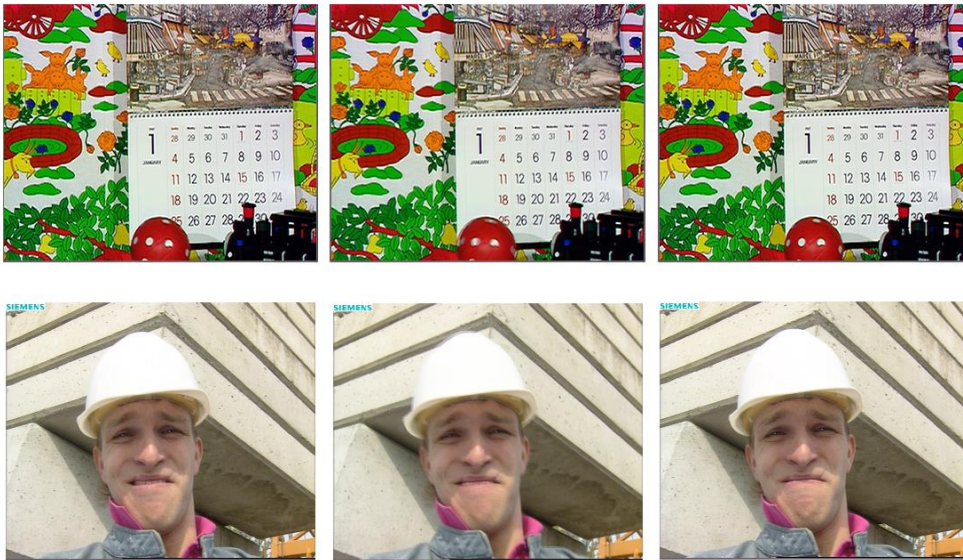


Figure 10. Results Of The Proposed System

Table 3. Performance Of The Implementation

Frequency of system clock	200MHz
Internal Memory	59.8KByte
Frame Size	1280×720
Frame Rate(f/s)	60
Throughput(pixels/s)	52.73M

Through the above analysis, the frame rate of image videos is upgrade from 60Hz to 120Hz by the proposed system based on bi-directional motion estimation, and the effect of the interpolated frames is similar to the original frames.

5. Conclusion

This paper proposed a real-time FPGA architecture of bi-directional motion estimation for FRUC system, which is based on processing element arrays. This architecture can effectively avoid holes and overlaps caused by unidirectional motion estimation. In this architecture, data multiplex and parallel processing techniques are fully used to reduce the computational complexity effectively. Meanwhile, pipelining technology is used to improve the system operating frequency. Experimental result shows that the architecture can real-time estimate accurate motion vectors under the 200MHz clock frequency. The architecture is easy for hardware implementation and can be used for video post-processing system.

Acknowledgment

This project is partially supported by the National Natural Science Foundation of China (Grant No. 61201347). The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers.

References

- [1] Lee Y L, Nguyen T, "High frame rate motion compensated frame interpolation in high-definition video processing", in Acoustics Speech and Signal Processing, 2010 IEEE International Conference on, pp. 858-861, (2010).
- [2] Byeong-Doo, C., et al., "Motion-Compensated Frame Interpolation Using Bilateral Motion Estimation and Adaptive Overlapped Block Motion Compensation", IEEE Transactions on Circuits and Systems for Video Technology, vol. 17, no. 4, pp. 407-416, (2007).
- [3] Gan P, Li Z, et al., "An Improved Block Motion Estimation Algorithm for FRUC System", Journal of Computational Information Systems, vol. 10, no. 4, pp. 1629-1637, (2014).
- [4] Truong Quang, V., P. Seok-Hwi, and K. Young-Chul, "Efficient architecture for hierarchical bidirectional motion estimation in frame rate up-conversion applications", in Computational Intelligence and Computing Research, 2010 IEEE International Conference on, (2010).
- [5] Li-Chang, L., et al, "A frame-level FSBM motion estimation architecture with large search range", in Advanced Video and Signal Based Surveillance, 2003 Proceedings. IEEE Conference on, (2003).
- [6] Hangu Y. and H. Yu-Hen, "A novel modular systolic array architecture for full-search block matching motion estimation", in Acoustics Speech and Signal Processing, 1995 International Conference on, (1995).
- [7] Yeong-Kang, L. and C. Liang-Gee, "A data-interlacing architecture with two-dimensional data-reuse for full-search block-matching algorithm", Circuits and Systems for Video Technology, IEEE Transactions on, vol. 8, no. 2, pp. 124-127,(1998).
- [8] Bo-Won, J., et al., "Coarse-to-fine frame interpolation for frame rate up-conversion using pyramid structure", IEEE Transactions on Consumer Electronics, vol. 49, no. 3, pp. 499-508, (2003).
- [9] Xuan Jing and Chau L.P, "An efficient three-step search algorithm for block motion estimation", IEEE Transactions on Multimedia, vol. 6, no. 3, pp. 435-438, (2004).
- [10] Dhahri S., Zitouni A. and Tourki R, "A parallel processing architecture for FSS block-matching motion estimation", in Communications, Computing and Control Applications, 2011 International Conference on, pp. 1-5, (2011).
- [11] Chun-Ho Cheung and Lai-Man Po, "A novel cross-diamond search algorithm for fast block motion estimation", Circuits and Systems for Video Technology, IEEE Transactions on, vol. 12, no. 12, pp. 1168-1177, (2002).
- [12] Wang, Y.-n. and Y.-y. Fan, "An Adaptive Diamond Searching Algorithm on Space-time Correlation", in Image and Graphics, 2007 Fourth International Conference on, (2007).

Author



Huangyang Fan, Professor, Ph.D. graduated from Dept. of Radio Engineering, Chongqing University, P.R. China, received bachelor's, master's and doctoral degrees respectively. He has been engaged in electronic technology, signal processing technology research and teaching. He has been teaching the course of the high frequency electronic circuit, TV technology, signal detection and estimation for undergraduate and graduate. Won the national teaching achievement award 1, Chongqing teaching achievement award 2, Chongqing University teaching achievement award 4; editor of books 1. As the person in charge of the project for the provincial and ministerial level scientific research, military project and the project more than 10; presided over the completion of Chongqing city and Chongqing University teaching reform project 5. Published more than 10 papers, won 4 national patents right. Subject: signal and information processing