

## Improving Energy Efficiency of Block-Matching Motion Estimation Using Dynamic Partial Reconfiguration

Jooheung Lee<sup>1</sup> and Jungwon Cho<sup>2,\*</sup>

<sup>1</sup> Dept. of Electronic and Electrical Engineering, Hongik University, Republic of Korea  
joolee@hongik.ac.kr

<sup>2</sup> Dept. of Computer Education, Jeju National University, Republic of Korea  
jwcho@jejunu.ac.kr

**Abstract.** In this paper, energy-efficient architecture for Variable Block Size Motion Estimation (VBSME) is proposed to fully utilize dynamic partial reconfiguration capability of programmable hardware fabric in the heterogeneous computing environment. Dynamic Partial Reconfiguration is a unique feature of FPGAs that makes best use of hardware resources and power by allowing adaptive algorithm to be implemented on the reconfigurable hardware fabric during runtime. We exploit this feature of FPGA to support run-time reconfiguration of the proposed modular hardware architecture for motion estimation (ME). The implemented scalable SAD array can provide different resolutions and frame rates for real time applications with multiple reconfigurable regions at the operating frequency of 90MHz.

**Keywords:** Variable Block Size Motion Estimation, Partial Reconfiguration.

### 1 Introduction

Due to the proliferation of heterogeneous embedded systems in the Internet of Things era, low complexity encoding for video sensing applications becomes essential especially in power-constrained environments. Current video encoding algorithms require high computing capability to achieve high compression ratios, but their computational complexities increase power consumption proportionally.

Motion estimation techniques play a key role in inter-video coding by removing temporal redundancies among video frames obtained from the conventional CCD/CMOS sensors [1-3]. As in H.264/AVC and High Efficiency Video Coding (HEVC) standards, motion estimation engine demands large amount of complex computational capabilities due to its support for wide range of different block sizes in order to increase accuracy in finding best matching block.

Recently, System-on-Chip (SoC) used for embedded video-sensing systems integrates various functional elements, such as general purpose processors, DSPs, customized hardware accelerators, memories, and programmable hardware fabrics (FPGAs). In this paper, we propose an approach to gain algorithmic and architectural

---

\* Corresponding Author



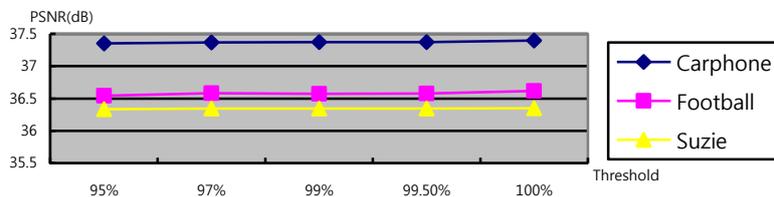
To perform block-based motion estimation, the matching criterion such as SADs in (1) among the current image block and all candidate blocks in the search range of reference frame is computed. Let the block size be  $N \times N$  and location of each block in the current frame (C) is represented by  $(i, j)$ , and a block within the search window  $(h, v)$  in the reference frame (R) is matched. Here,  $c(x,y)$  is a pixel value in the current block and  $s_w(x,y)$  is a search candidate block. The center  $(x1, y1)$  of the search range is predicted, depending on the behavior of motion vectors of neighboring blocks.

### 3 Experimental Results

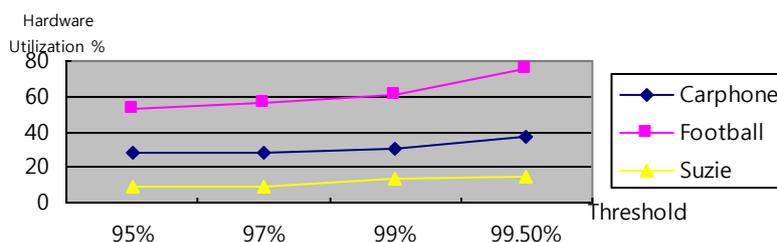
Performing exhaustive search is computationally very expensive and adds great complexity to motion estimation. A full search algorithm for each block candidate requires  $(2R+1)^2$  SAD computations where  $[-R,+R]$  is the search window size. We reduce this search range (R) adaptively based on the correlation of the neighboring blocks' motion vectors. The adaptive search range reduction for dynamic partial reconfiguration of modular motion estimation HW architecture is described below.

- a) Initially set the search range for all the reference frames (for the case of multiple or single referenced frame algorithm) to maximum search range size (i.e.,  $\pm 32$ ).
- b) Perform prediction algorithm as used in the JM software and obtain Motion Vector Prediction (MVP). Then, set the search range center in the reference frame accordingly. For simplicity in hardware implementation, all the block types for current macroblock share the same MVP (Fast Full Search in JM).
- c) Perform Fast Full Search motion estimation and simultaneously store the number of MVs falling into different category of absolute displacements. (i.e., number of MVs with displacement 0, 1, 2, and so on).
- d) Set the search range for the next frame such that the search range covers at least Th (Threshold) % of total MVs. 'Th' can vary from 90-100% depending on the user requirement: higher performance/PSNR, less computational complexity, or hardware resources utilized, and so on. In the proposed approach, search range is updated at frame level due to the overhead of bitstream loading from the external memory for real-time reconfiguration on the FPGA device. Hence, it is assumed that the search range is fixed for all macroblocks in a particular frame.
- e) Continue from step (b) onwards.

The performance comparison of Search Range Reduction Algorithm with the JM 12.4 Reference Software is given in Fig. 2. Threshold values from 95% to 99.5% are used for simulations on various video sequences (QCIF/30fps), and their corresponding results are shown in terms of PSNR and hardware utilization. For comparisons, Fast Full Search algorithm using the same search range of 32 for all blocks of the given macroblock is used for encoding 60 frames. Parameters in JM software are set to baseline profile (no B slices), one reference frame, and fixed search range. As shown in Fig. 2, the proposed adaptive search range approach using partial reconfiguration is very useful in reducing the computational complexity and hardware requirement significantly at the cost of little degradation in video quality, i.e., small increase in bitrate and decrease in PSNR.



(a) P SNR vs. Threshold



(b) Hardware Utilization vs. Threshold

Fig. 2. Performance comparison of adaptive SR with partial reconfiguration

## 4 Conclusion

In this paper, we presented the design of motion estimation processing engine which is capable of executing for different search range values. The proposed motion estimation architecture is suitable for real-time applications where hardware resource is a critical criterion. First, a search range reduction is introduced at frame level, in order to intelligently adapt towards various video sequences and at the same time the computational complexity can be reduced further wherever possible with only small degradation in PSNR. Second, modular hardware architecture is presented for ME engine to process with different SR values during run time.

## References

1. Dias, T., Momcilovic, S., Roma, N., and Sousa, L. In: Adaptive motion estimation processor for autonomous video devices. EURASIP Journal on Embedded Systems, pp. 1--10. (2007)
2. Song, Y., Liu, Z., Goto, S., and Ikenaga, T. In: Scalable VLSI Architecture for Variable Block Size Integer Motion Estimation in H.264/AVC. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, pp. 979--988. (2006)
3. Liu, Z., Song, Y., Ikenaga, T., and Goto, S. In: A fine-grain scalable and low memory cost variable block size motion estimation architecture for H.264/AVC. IEICE Transactions on Electronics, pp. 1928--1936. (2006)

4. Gupta, S., and Lee, J. In: A Scalable H.264/AVC Variable Block Size Motion Estimation Engine Using Partial Reconfiguration. Proceedings of International Conference on Engineering of Reconfigurable Systems and Algorithms, (2009)