HIGH PERFORMANCE HARDWARE ARCHITECTURE FOR CONSTRAINED ONE-BIT TRANSFORM BASED MOTION ESTIMATION

Anıl Çelebi, Oğuzhan Urhan

Electronics & Telecom. Eng. Dept., University of Kocaeli
Kocaeli , Turkey
phone: + (90) 262 303 33 54, fax: + (90) 262 303 33 49, email: anilcelebi@kocaeli.edu.tr
phone: + (90) 262 303 33 52, fax: + (90) 262 303 33 49, email: urhano@kocaeli.edu.tr
web: http://ehm.kocaeli.edu.tr

ABSTRACT

Motion estimation (ME) processes is considered as the most computationally intensive part of the conventional video compression standards. Low bit-depth representation based ME approaches provides an important alternative to reduce this computational load by making use of a lightweight and hardware efficient matching criteria. Constrained one-bit transform (C-1BT) based ME employs only two bit-planes and stands out its superior performance compared to other low bit-depth based ME approaches. Recently an adaptive search range determination algorithm is proposed to further speed-up C-1BT based ME. This paper presents novel hardware architecture for adaptive search range determination based ME method mentioned above. Proposed architecture implements spiral search method. No on-chip memory is needed neither for reference and nor for current macroblocks. Thus, there is no need to design a complex memory hierarchy and control logic to implement the spiral search method. A data reuse scheme among adjacent search windows is utilised with thanks to two axis rotatable two dimensional shift register architecture. Thus, a very low off-chip memory bandwidth can be achieved.

1. INTRODUCTION

Many devices such as camcorders, mobile phones have video capturing capabilities. Because it is not feasible to store captured image frames in raw form. Efficient compression methods and standards are used to effectively reduce data amount to be stored and/or transmitted. Temporal redundancy between image frames is exploited using motion estimation (ME) approaches. However, block based ME processes consumes up to 80% of the total computational load in a typical video encoder [1]. Thus, many researchers focus on reducing computational complexity of ME without significantly degrading ME accuracy.

In block based ME (BME) approach non-overlapping image blocks are searched in the reference frame within a predefined search range. When all possible search centres are used in this matching process it is called as full search based ME (FSBME). There are many approaches to reduce computational complexity of FSBME in the literature. These approaches can be divided into several categories. The first group employs reduced number of search centre. Three step search (3SS) based ME presented in [2] makes use of a total

of 27 search centre. Thus, total computational load of FSBME is significantly reduced at the cost of limited ME accuracy. The approach in [3], utilizes a hexagonal shaped search pattern to control only limited number of search centres at a reduced computational cost compared to 3SS based ME. Another group of methods take only certain number of pixels into account in the computation of matching criteria using certain sub-sampling pattern. Thus, number of computations to compute motion vector can be reduced. For example, an N-Queen lattice structure is utilized to reduce number of pixel to be used in the matching process in [4]. Early termination based methods aim to finalize search process in a very early step to reduce computational load. In [5], search centre prediction in combination with an early termination approach is presented. This method claims up to 160 times speed-up compared to FSBME for sequences containing very low motion activity. Another group targets to change the pre-defined search range adaptively for each block as in [6].

The last group propose to use lightweight and hardware efficient matching criterion [7-12]. In [7], video frames are initially converted to binary images using multi band-pass filtered video frames as adaptive threshold. Next, a Boolean EX-OR operation based matching criteria is utilized. This method is called as one-bit transform (1BT) based ME. In [8], an additional bit-plane is derived using local image features. Then, two bit-planes for each frame are employed together to compute matching criteria. This method is known as two-bit transform (2BT) based ME and has higher accuracy compared to 1BT based ME. However, computational load of 2BT is higher than 1BT. In [9], multiplication operations at multi band-pass filtering stage are omitted. This, so called multiplication-free one-bit transform (MF-1BT) based approach provides similar ME accuracy compared to 1BT, resulting in a lower transform cost. In [10], a constraint mask (CM) is created to discriminate *reliable pixels* in 1BT based ME. The CM and 1BT bit planes are employed together to compute matching criterion. Additional cost of this method (C-1BT) is quite low compared to 2BT. Furthermore, it provides better performance than 1BT and 2BT based ME methods. In [11], hardware architecture for C-1BT method is presented first in the literature. Recently, truncated versions of Gray-coded bit-planes are employed in the matching process in [12]. This approach makes

use of three bit-planes and provides better performance than previously proposed low-bit depth methods. Several novel architectures that clearly show the effectiveness of the low-bit depth pixel representation based ME methods are presented in [12-16].

Recently, there have been attempts to further reduce the computational load of low-bit depth based ME approaches. In [17], diamond search and 1BT based ME is combined to further speed-up 1BT based ME. However, motion vector accuracy is considerably degraded in this case. In [18], a predictive hexagonal search approach and partial distortion search method is combined with C-1BT based ME and it is shown that the significant reduction on computational load is possible with small amount of performance loss. In [19, 20] special early termination approaches are utilized for 1BT and 2BT based ME. Combination of adaptive search range with low bit-depth methods is recently investigated in [21]. This method initially determines the search range for each block using a simple computation then, motion estimation is performed. Experiments have been show that this approach can provide up to 90% gain in computational load. Note that although the method presented in [18] provides significant computational gain, its irregular data access may hinder efficient hardware implementation. However the method in [21] has regular data access and this enable efficient hardware implementation.

In this paper we present an efficient hardware architecture method proposed in [21]. Proposed architecture does not need dedicated on chip memory since a D-type flip flop (DFF) based two axis rotatable two dimensional register array is utilised to implement both a shifting engine and memory module at the same time. Thus, the control logic of whole architecture is reduced and complex data read schemes are eliminated with thanks to sequential row-column read operations is eliminated.

2. C-1BT BASED ME USING ADAPTIVE SEARCH RANGE

Constrained one-bit transform based (C-1BT) based ME approach makes use of two bit planes. The first bit plane is created by simply filtering the video frames with a diamond shaped kernel proposed in [9] and then comparing the filtered frames against to original video frames. This operation is formulated as follow:

$$B(i,j) = \begin{cases} 1, & \text{if } I(i,j) \ge I_F(i,j) \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

$$CM(i,j) = \begin{cases} 1, & \text{if } |I(i,j) - I_F(i,j)| \ge D \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

where I and I_F show original and filtered image frames, respectively. The second bit plane which corresponds to constraint mask (CM) is evaluated by subtraction operation between the respective pixels of original and filtered video frames. The pixels that have higher difference are considered as reliable. The computation of CM is given in (2).

D is fixed to 10 in [10]. The matching criteria used in C-1BT based ME is called as constrained number of non-matching points (CNNMP) and it is given as follows:

$$CNNMP(mx,my) = \sum_{i=0}^{N-1}\sum_{i=0}^{N-1} \left\{ \begin{array}{l} \left[CM^t(i,j) \,\|\, CM^{t-1}(i+mx,j+my) \right] \\ \bullet \left[B^t(i,j) \oplus B^{t-1}(i+mx,j+my) \right] \end{array} \right\} \quad (3)$$

where $\|, \bullet,$ and, $\oplus$ denote binary OR, AND, and EXOR operations, respectively. The location (mx,my) giving the lowest matching error is considered as the motion vector for the current block. In [21], one-bit images of C-1BT based ME is employed to decide search range for each block. This approach is based on the block activity measurement. If a block has low activity then, a small search range will probably provide enough motion vector accuracy. On the other hand, if the motion activity is high then, a larger search area should be employed. Based on this discussion, the search range (SR) for each block in C-1BT is decided as follows in [21]:

$$SR = \frac{\sum_{i=0}^{N-1}\sum_{i=0}^{N-1}\left[B^t(i,j) \oplus B^{t-1}(i,j) \right]}{\alpha} + \beta \quad (4)$$

where α and β are fixed to 12 and 2, respectively. In this paper, a modified version of (4) is proposed and utilised to make the computation easier as follow:

$$SR\_\text{mod} = \frac{\gamma \times \sum_{i=0}^{N-1}\sum_{i=0}^{N-1}\left[B^t(i,j) \oplus B^{t-1}(i,j) \right]}{\alpha'} + \beta' \quad (5)$$

where γ, α' and β' is fixed to 3, 32 and 1, respectively. Thus, 1 bit right shift, 10 bits addition and 5 bits left shift is needed. Neither arithmetic multiplication nor arithmetic division is needed. Note that β' is not needed for hardware implementation since the default search range is arranged as [1,-1] in proposed architecture. Note that all these operations can be performed using only integer arithmetic. Maximum search range is limited to [-16, 16] by a simple magnitude comparison on hardware.

According to the experimental results given in Table 1 there is only a slight difference (0.01dB fall in average) in ME accuracy caused by the change performed on (4). On the other hand, proposed modified search range determination approach provides 1% computational gain compared to [21]. Note that average of search range (Av. of SR) is also provided in this table.

3. PROPOSED HARDWARE ARCHITERTURE FOR C-1BT WITH ADAPTIVE SEARCH RANGE

In this paper, novel hardware architecture for C-1BT based ME with adaptive search range method is proposed. Additionally, spiral search scheme is utilised which is not mentioned in [21]. The block diagram of the data path of the hardware architecture is shown in Figure 1.
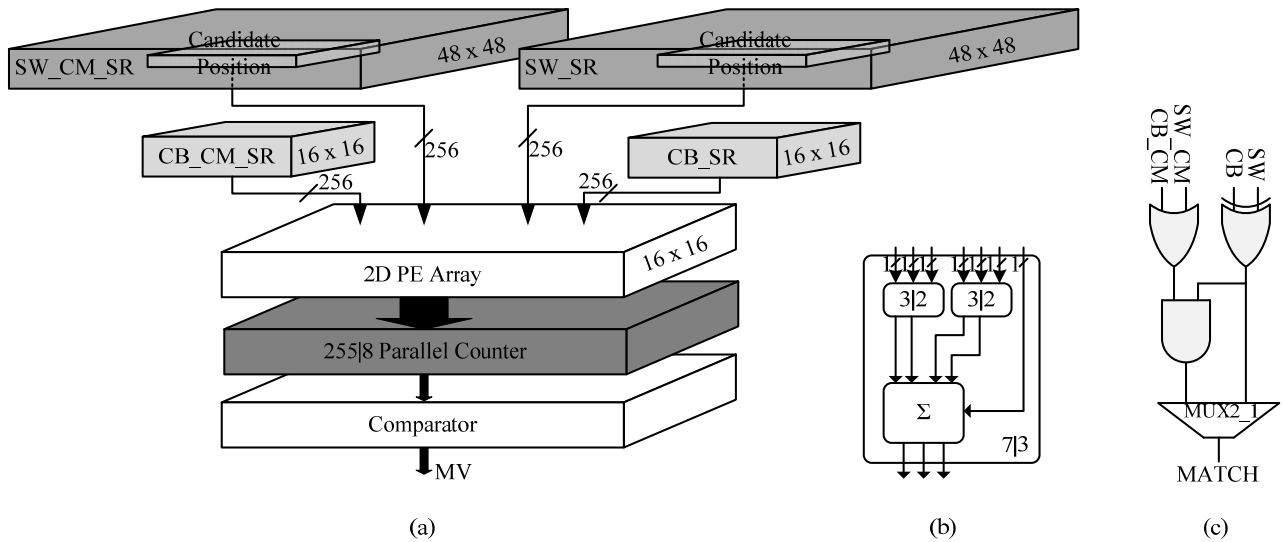
Figure 1 – a) Proposed Hardware Architecture, b) Basic datapath for parallel counter, c) Processing element(PE)

Table 1. ME Performance Comparison.

Sequence	The method presented in [21]			Proposed Method		
	Gain (%)	Av. of SR	PSNR (dB)	Gain (%)	Av. of SR	PSNR (dB)
Football	78,91	7,35	22,06	79,92	7,17	22,06
Foreman	79,26	7,29	30,71	80,16	7,13	30,69
Tennis	78,75	7,38	28,60	79,41	7,26	28,59
Garden	69,51	8,83	23,44	69,98	8,77	23,43
Mobile	90,03	5,05	23,67	91,90	4,55	23,69
Coastguard	80,80	7,01	30,05	82,22	6,75	30,00
Average	**79,54**	**7,15**	**26,42**	**80,60**	**6,94**	**26,41**

In Figure 1a, there are two 16×16 bits shift registers for current block, two 48×48 shift registers for search window where the second shift registers utilised for the constraint mask. Current block shift registers are capable of shifting to the right or down according to the moving direction for loading the next macroblock. On the other hand the shift register array for the search window is capable of moving the data to 4 directions of right, left, up and, down with rotate capability that enables the proposed spiral search mechanism. The architecture of the search window shift register is shown in Figure 2.

The spiral search scheme is implemented with thanks to the search window shift register. These modules contain 48×48 DFFs both but only have 16×16 outputs which are located on the centre of the register array. To perform a move to the next search location on the right, the register array is configured to shift the data to the left to generate the relative move to right. Also note that the bits on the boundaries of the movement are concatenated to the input of the registers on the opposite end of the array. At the end of the search process none of the pixel information in the PE array is lost that allows reuse of the data for the next macroblock that the motion vector of which is going to be computed. In Figure 3, the spiral search scheme for a 2×2 block and search range of [-2, 2] illustrated. The numbers in the squares represent the next position of the dark grey pixel located on the bottom right corner of the block for each search step.

The 2D PE array and one PE component is shown in Figure 1a and 1c respectively. The PE architecture contains an additional 2×1 multiplexer to enable search range computation and CNNMP computation selectively. 16×16 PEs are utilised to implement the equation (5).

The parallel counter block which counts the non-matching pixels of a candidate search position and the sub parallel counter block which counts the number of non matching point criterion for 7 pixels are shown in Figure 1a and 1b respectively. The architecture of parallel counter consists of seven stages of sub parallel counters of size $3|2$, $7|3$, $15|4$, $31|5$, $63|6$, $128|7$ and, $255|8$ respectively. There are 256 pixels in each macroblock but parallel counter has 255 inputs. Experimental results have shown that the absence of one pixel in the CNNMP computation does not affect the ME performance there for one stage of addition for only one pixel is neglected to reduce the complexity.
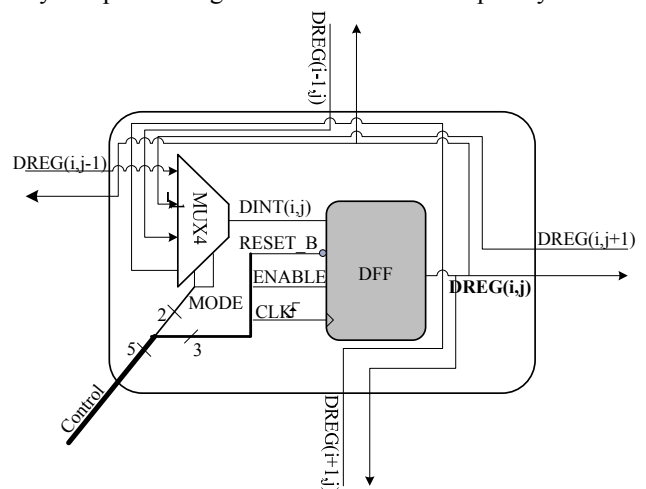

Figure 2- Architecture of search window shift register.

19 18 17 16 15
20 6 5 4 14
21 7 0 3 13
22 8 1 2 12
23 24 9 10 11

Shift register rotates upwards
Shift register rotates left
Shift register rotates downwards
Shift register rotates right

Figure 3- Utilized spiral search scheme for the proposed hardware architecture.

The combinational path delay of the parallel counter block has been obtained about 11,57ns for a 45nm technology FPGA device. Actually this number reduces the maximum achievable clock frequency to approximately 90 MHz that is why 2 pipeline stages are added between the sub parallel counter stages of $15|4-31|5$ and $63|6-127|7$. The output of the parallel counter is also synchronised with clock to divide the combinational path to next stage where comparison operation is performed.

In [7] an 8 input LUT based architecture is proposed to count the number of non-matching pixels. Then, several architectures are proposed to overcome the bottleneck due to the logarithmic relation between input width and the LUT depth. In [14] a 4 input LUT based non-match counter architecture is proposed to reduce the area of the hardware. Recently in [16], parallel counter method is proposed to count the number of non-matching pixels and according to the best of our knowledge it is the most hardware efficient method reported yet.

4. IMPLEMENTATION RESULTS

Proposed hardware architecture is implemented on a state of the art FPGA that is fabricated on 45nm process. According to the synthesis results proposed hardware architecture occupied 5691 LUTs and 5309 DFFs which consume the 6% and 2% of the total available resources of the FPGA device used, respectively. According to the post place and route timing analysis results, the proposed hardware architecture can operate up to 197MHz clock frequency.

There is not any fixed number of clock cycles that the hardware architecture needs for computing the best candidate position of a macroblock. But for the worst case where the search range is [-16,16]. Total number of candidate position is 1089. Other than that the hardware architecture needs 4 clock cycles for computing search range for a macroblock because of the pipelined architecture structure of the parallel counter shown in Figure 1a and 1b. Then additional 5 clock cycles are needed at the end of the computation for control purpose. At the end of the search process for the range of [-16, 16] an additional 16 clock cycles are needed to recover the search window to its initial phase. Because, the next search window is going to be concatenated to the current search window. In Figure 3, it is shown that with thanks to the utilised spiral search scheme one pe-

riod of the search step around the square ends exactly one step below the previous location where the first starting point is (0,0) position. Thus a rotate shift operation to the up is enough to recover the search window to its initial state. Consequently 1114 clock cycles are needed for the worst case. But the total workload of the dynamic search range is not same for each macroblock. According to the results given in Table 1 the computational load of the C-1BT based ME with adaptive search range can be lowered up to 90% compared to the full search which means, total number of clock cycles needed for a macroblock changes between 1114 and 120 approximately for the proposed hardware architecture. A comparison of the performance between several hardware architectures proposed in the literature are shown in Table 2.

5. CONCLUSIONS

Novel hardware architecture for one of the most recent low complexity ME methods in the literature is proposed. With thanks to the adopted ME method a hardware architecture which has the potential of processing a very high resolution video sequence on very high frame rate is obtained. Such that, according to Table 1 average search range search range for a video sequence is around 7. Since additional 9 control cycles are needed for each macroblock, an average clock cycle of $7+9+(2*7+1)^2=241$ is needed. For the clock frequency of 197MHz obtained after post place and route timing analysis, a processing power of 847k macroblocks/second can be achieved. Thus, approximately 101 frames/second performance for 1080p resolution is achieved that is superior compared to the architecture proposed for single reference frame operation in recently published [15].

Table 2. ME Performance Comparison.

	Proposed	[15]	[11]
Bit Depth	2	1	1
On Chip Memory	0	4096	24064
Area	5691 LUTs 5309 DFFs	3914 LUTs 2517 DFFs	1467 LUTs 499 DFFs
Maximum Frequency	197 MHz	192 MHz	127 MHz
Technology	FPGA 45nm	FPGA 65nm	FPGA 65nm
Search Range	Dynamic From [-1,1] to [-16,16]	[-16, 16]	[-16, 16]
Search Method	Spiral Search	Full Search	Full Search
1080p Performance	101 fps	84 fps	15 fps
720p Performance	225 fps	189 fps	33 fps
# of Reference Frames	1	1	1

REFERENCES

[1] P.M. Kuhn, *Algorithms, Complexity Analysis and VLSI Architectures for MPEG-4 Motion Estimation*, Boston, MA: Kluwer Academic, 1999.

[2] T. Koga, K. Iinuma, A. Hirano, Y. Iijima, and T. Ishiguro, "Motion compensated interframe coding for video conferencing," *in: Proceedings of the National Telecommunications Conference*, New Orleans, November 29–December 3. 1981,pp. G5.3.1–G5.3.5.

[3] S. Zhu and K.K. Ma, "A New Diamond Search Algorithm for Fast Block-Matching Motion Estimation," *IEEE Trans. Image Process.*, vol. 9, pp. 287-290, 2000.

[4] C.-N. Wang, S.-W. Yang, C.-M. Liu, and T. Chiang, "A Hierarchical N-Queen Decimation Lattice and Hardware Architecture for Motion Estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, pp. 429-449, 2004.

[5] C. Shen, S. Li, "Search centre prediction-based partial distortion search motion estimation algorithm," *Electronics Letters*, vol. 43, pp. 213-214, 2007.

[6] S. Goel, Y. Ismail, and M. A. Bayoumi, "Adaptive search window size algorithm for fast motion estimation in H.264/AVC standard," *in Proc. of the Midwest Symp. on Circuits and Systems 2005*, Cincinati, Ohio, USA, August 7-10. 2005, pp. 1557-1560.

[7] B. Natarajan, V. Bhaskaran, and K. Konstantinides, "Low-complexity block-based motion estimation via one-bit transforms," *IEEE Trans. Circuit Syst. Video Technol.*, vol. 7, pp. 702-706, 1997.

[8] A. Ertürk and S. Ertürk, "Two-Bit Transform for Binary Block Motion Estimation," *IEEE Trans. Circuit Syst. Video Technol.*, vol. 15, pp. 938- 946, 2005.

[9] S. Ertürk, "Multiplication-free one-bit transform for low-complexity block-based motion estimation," *IEEE Signal Process. Lett.*, vol. 14, pp. 109-112, 2007.

[10] O. Urhan and S. Ertürk, "Constrained one-bit transform for low-complexity block motion estimation," *IEEE Trans. Circuits and Syst. Video Technol.*, vol. 17, pp. 478-482, April 2007.

[11] A. Çelebi, O. Urhan, I. Hamzaoğlu, S. Ertürk, "Efficient Hardware Implementations of Low Bit Depth Motion Estimation Algorithms," *IEEE Signal Process. Letts.*, vol. 16, pp. 513-516,2009.

[12] A. Çelebi, O. Akbulut, O. Urhan, and S. Ertürk, "Truncated Gray-Coded Bit-Plane Matching Based Motion Estimation and its Hardware Architecture," *IEEE Trans. Consumer Electron.*, vol. 55, pp. 1530-1536, 2009.

[13] A. Çelebi, O. Akbulut, O. Urhan, I. Hamzaoğlu, S. Ertürk, "An All Binary Sub-Pixel Motion Estimation Approach and its Hardware Architecture," *IEEE Trans. Consumer Electron.*, vol. 54, no. 4, 2008.

[14] A. Akın, Y. Doğan, İ. Hamzaoğlu, "High Performance Hardware Architectures for One Bit Transform Based Motion Estimation," *IEEE Trans. Consumer Electron.*, vol. 55, pp. 941-949, 2009.

[15] A. Akın, G. Sayılar, I. Hamzaoğlu, "High Performance Hardware Architectures for One Bit Transform Based Single and Multiple Reference Frame Motion Estimation," *IEEE Trans. Consumer Electron.*, vol. 56, pp. 1144-1152, 2010.

[16] A. Çelebi, H. J. Lee, S. Ertürk, "Bit Plane Matching Based Variable Block Size Motion Estimation Method and Its Hardware Architecture," *IEEE Trans. Consumer Electron.*, vol. 56, pp. 1625-1633, 2010

[17] E.S. Lee, O. Urhan, T.G. Chang, "Multiplication-free one-bit transform and diamond search combination for fast binary block motion estimation," *in: Proc. IEEE 15th Conf. on Signal Processing and Communications Applications*, Eskişehir, Turkey, June 11-13. 2007, pp. 430-433.

[18] O. Urhan, "Constrained one-bit transform based motion estimation using predictive hexagonal pattern," *Journal of Electron. Imaging*, vol. 61, Article ID: 033019, 2007.

[19] H. Lee, J. Jeong, "Early termination scheme for binary block motion estimation," *IEEE Trans. Consumer Electron.*, vol. 53, pp. 1682-1686, 2007.

[20] H. Lee, S. Jin, J. Jeong, "Early termination algorithm for 2BT block motion estimation," *Electronics Lett.*, vol. 45, pp. 403-405, 2009.

[21] O. Urhan, "Constrained One-Bit Transform Based Fast Block Motion Estimation using Adaptive Search Range," *IEEE Trans. Consumer Electron.*, vol. 56, pp. 1868-1871, 2010.