Successive elimination algorithm for truncated gray-coded bitplane matching based motion estimation

Ilseung Kim, and Jechang Jeong

Abstract—In this paper, a successive elimination algorithm for truncated gray-coded bitplane matching (TGCBPM) based motion estimation (ME) is proposed. After calculating the lower bound for TGCBPM, we can substantially reduce the computational complexity by skipping the impossible candidate. Experimental results show that this proposed algorithm has an outstanding performance in terms of the computational complexity although the motion accuracy of the proposed algorithm is the same as that of the full search TGCBPM.

Keywords— Video coding, motion estimation, full-search, truncated gray-coded bit-plane matching (TGCBPM)

I. Introduction

Motion estimation (ME) plays a critical role in the video compression process, since it considerably contributes coding efficiency by reducing possible temporal redundancies between frames. The full search-based sum of absolute differences (FS-SAD) or sum of squared differences (FS- SSD) methods can be the optimal estimation of the motion, which conducts exhaustive searches within its search range in order to find the minimum matching error based on sum of absolute differences. However, the FS-SAD method is not suitable for real-time applications because it requires excessive computation.

In order to alleviate this problem, many approaches have been proposed. Most of the techniques try to minimize the whole calculations of the matching criterion [1]-[4] or searching points and successive elimination algorithm (SEA) is one of those [5]. In SEA, if the pre-calculated lower bound of the block matching error is larger than the up-to-date minimum distortion, the block matching process of the block will be skipped safely. With this process, a number of the matching computations are reduced, while the ME accuracy of the SEA is the same as that of the full search algorithm.

Some algorithms have been proposed with the different approaches that utilize different matching criteria from the conventional SAD or SSD, which exploit Boolean operations in order to speed up the calculation of the matching criterion itself [6]-[10]. These techniques are called binary block motion estimation or bit-plane matching (BPM)-based MEs. These techniques have two main advantages compared to the typical ME algorithms based on SAD or SSD: fast computation of the matching criterion using Boolean operation and reduced memory bandwidth.

In [6], bit planes for motion estimation are generated by comparing the original image frame against the multi-bandpass filtered frame, whose method is called one-bit transform (1BT)-based ME. In [7], a multiplication-free 1BT (MF1BT) is proposed, which exploits a multiplication-free bandpass filter kernel in order to generate bit planes. Two-bit transform (2BT) and constrained 1BT (C-1BT) were suggested to improve the motion accuracy of 1BT-based ME approaches [8], [9]. In [8], two bit-planes are constructed employing basic statistical information such as mean and standard deviation. In [9], a bit-plane, called a constrained bit-plane, is added to refine the performance of 1BT-based ME. In [10], the truncated gray coded bit-plane matching (TGVBPM) based ME was proposed. This method switches the original frames' pixels into gray-coded ones, chooses some bit planes from the most significant bit (MSB), and conducts the block matching process using the chosen bit planes. The authors of [10] recommend that 3 bit-planes are good enough to improve performance of previously presented BPM-based ME methods.

BPM-based ME methods take advantage of fast computation of their criteria, but they still exploit FS strategy for all possible candidate in the search range. By using the fact, several algorithms were proposed to further reduce the computational complexity of the BPM-based ME algorithm by exploiting early termination scheme [11]-[12] and combining search range adjustment method [13]. Note that the ME accuracy of those algorithms are not the same as that of the FS-BPM-based ME methods.

SEA method is one of the techniques which effectively reduces the complexity of BPM-based ME without any loss [14] – [16]. The SEA for 1BT was derived based on the triangle

This research was supported by the MSIP(Ministry of Science, ICT & Future Planning), Korea, under the project for technical development of information communication & broadcasting supervised by IITP(Institute for Information & Communications Technology Promotion)(IITP 2015-B0101-15-1377).

Ilseung Kim is with Hanyang University, Seoul, Korea. He is now with the Department of Electronics and Computer Engineering, Seoul, Korea. (e-mail: ghanjang@gmail.com).

Jechang Jeong is with Hanyang University, Seoul, Korea. He is now with the Department of Electronics and Computer Engineering, Seoul, Korea. (phone: 82-2-2220-4369; e-mail: jjeong@hanyang.ac.kr).

inequality to remove the impossible candidates in [14]. In [15], the authors derived the lower bound of the block matching error of 2BT by using set theory and triangle inequality. Recently, SEA for C1BT was proposed [16]. Because the matching criterion of C1BT does not satisfy the metric conditions, the authors derive the lower bound of the block matching error using the Bonferroni inequality.

In this paper, SEA for TGCBPM is proposed. The proposed algorithm analyzes the TGCBPM matching criterion and derives the lower bounds of it based on mathematical techniques in order to discard the impossible candidates and save computations significantly. Experimental results demonstate that the proposed algorithm reduces computational complexity substantially while maintaining the entirely same ME accuracy.

The rest of this paper is organized as follows. In section II, we review the previouse SEA techniques for 1BT and 2BT, and C1BT. Section III presents our proposed algorithm. Experimental results are provided in Section IV. The conclusion is set forth in Section V.

II. Previous Works

In this Section, the previous works of SEA algorithms for BPM based ME methods are reviewed including SEAs for 1BT and 2BT, because some results in previous works are some of the bases of the proposed algorithm.

A. Successive Elimination Algorithm for 1BT

The derivation of the SEA for 1BT in [14] starts from the basic triangle inequality for binary values. Let b_1 and b_2 be any two one-bit binary value, and then the following inequality holds:

$$|b_1 - b_2| \le b_1 \oplus b_2 \tag{1}$$

where $\oplus$ denotes the Boolean X-OR operation.

Streching the meaning of this result in terms of the matching measure of 1BT, i.e. $NNMP_{1BT}$, we can derive the following result:

$$\begin{aligned} NNMP_{1BT}(m,n) &= \sum_{i,j=0}^{N-1} \left| B^t(i,j) \oplus B^{t-1}(i+m, j+n) \right| \\ &\ge \sum_{i,j=0}^{N-1} \left| B^t(i,j) - B^{t-1}(i+m, j+n) \right| \\ &\ge \left| \sum_{i,j=0}^{N-1} B^t(i,j) - B^{t-1}(i+m, j+n) \right| \end{aligned} \tag{2}$$

where B^t denotes the one-bit representation of the current frame, B^{t-1} denotes that of the reference frame, (i, j) represent the search points in the search range [0, N-1], and (m, n) represent displacement of a motion vector, respectibly. If the sum norm of the block is not satisfied with (2), the search is not performed at the point. The calculation of the sum norm can be done efficiently by using the method in [5].

B. Successive Elimination Algorithm for 2BT

Let $\mathbf{B}$ be a vector as binary seqeunces $b(i)$ $(0 \le i \le N-1)$. The authors in [15] introduce the concept of the Hamming weight in order to concisely express the sum norm of the blocks and utilize the set theory. The relationship between sum norm of $b(i)$ and the Hamming weight of $\mathbf{B}$ can be driven with the following equality:

$$\sum_{i=0}^{N-1} b(i) = w_H(\mathbf{B}) \tag{3}$$

where $w_H(\cdot)$ denotes the Hamming weight which is the number of nonzero components.

By using the concept of the set theory, we obtain the following inequality:

$$w_H(\mathbf{x} \,\|\, \mathbf{y}) \ge \max\left(w_H(\mathbf{x}), w_H(\mathbf{y})\right) \tag{4}$$

where $\|$ denotes the Boolean OR operation and $\mathbf{x}$ and $\mathbf{y}$ represent arbitrary two vectors.

Matching criterion of 2BT is given as

$$\begin{aligned} NNMP_{2BT}(m,n) = \sum_{i,j=0}^{N-1} &\left\{ B_1^t(i,j) \oplus B_1^{t-1}(i+m, j+n) \right\} \\ &\| \ \left\{ B_2^t(i,j) \oplus B_2^{t-1}(i+m, j+n) \right\} \end{aligned} \tag{5}$$

where $B_{1,2}^t$ and $B_{1,2}^{t-1}$ are the two-bit representations of current and reference frames, (i, j) represents the search points in the search range [0, N-1], (m, n) represents displacement of a motion vector, the motion block size is $N \times N$ and $-s \le m, n \le s$ is the search range.

The result of the SEA for 2BT can be obtained by using the results of (2) and (4) as follows:

$$\begin{aligned} &NNMP_{2BT}(m,n) \\ &\ge \max\left(\left| w_H\left(\mathbf{B_1^t}\right) \right| - \left| w_H\left(\mathbf{B_{1,mn}^{t-1}}\right) \right|, \left| w_H\left(\mathbf{B_2^t}\right) \right| - \left| w_H\left(\mathbf{B_{2,mn}^{t-1}}\right) \right| \right) \end{aligned} \tag{6}$$

where we identify a vector $\mathbf{B_{1,2}^t}$ as $B_{1,2}^t(i,j)$ and a vector $\mathbf{B}_{\mathbf{1,2},mn}^{\mathbf{t-1}}$ as $B_{1,2}^{t-1}(i+m, j+n)$.

III. PROPOSED ALGORITHM

In this section, the derivation of the lower bound for TGCBPM matching criterion using the result in section II is presented. In order to derive the SEA for TGCBPM-based ME, the analysis of the TGCBPM-based ME algorithm is necessary.

A pixel value that is quantized to 2^K grey levels can be represented as:

$$f^t(x,y) = a_{K-1}2^{K-1} + a_{K-2}2^{K-2} + \cdots + a_1 2^1 + a_0 2^0 \quad (7)$$

where a_k coefficients represent the natural binary code and take only binary values and K denotes the bit-depth for the pixel. The gray-coded version of a pixel value can be obtained from its natural binary codes with the following equations:

$$\begin{cases} g_{K-1} = a_{K-1} \\ g_k = a_k \oplus a_{k+1} \end{cases}, 0 \le k \le K-2 \ . \quad (8)$$

The matching criterion for the TGCBPM-based ME, called the correlation metric CM_{TGC}, is given as

$$\begin{aligned} & CM_{TGC}(m,n) \\ & = \sum_{i,j=0}^{N-1} \sum_{k=NTB}^{K-1} 2^{K-NTB} \left\{ g_k^t(i,j) \oplus g_k^{t-1}(i+m, j+n) \right\} \\ & -s \le m,n \le s-1 \end{aligned} \quad (9)$$

where (m, n), s, k, NTB, and g_k^t denote the candidate displacement and search range, bit level, the number of truncated bit, and kth level of the binary pixel value with gray-coded version, respectively. TGCBPM based ME method makes only use of the highest ($K-NTB$) bit-planes to compute the matching process. Note that, higher order bit-planes have higher weight by a scaling factor of 2^{K-NTB}. The ME performance of [10] depends on NTB and the authors in [10] suggested the value of NTB as 5, which means only 3 bits from the MSB are used for the block matching process.

The purpose of SEA is that the pre-calculated lower bound of the block matching error is used to skip the impossible candidate that makes the ME process faster. The performance of SEA depends on how fast the lower bound is calculated. The algorithms in [14] – [16] make the fast computation of the lower bound of the matching error by using mathmatical techniques suggested in (1) ~ (6), which contributes to increaing the overall ME performance in terms of computational time.

However, it is not appropriate to directly apply the result from the previous works in case of TGCBPM. Applying the inequality (1) into the (7), we get the following inequality:

$$\begin{aligned} & CM_{TGC}(m,n) \\ & = \sum_{i,j=0}^{N-1} \sum_{k=NTB}^{K-1} 2^{K-NTB} \left\{ g_k^t(i,j) \oplus g_k^{t-1}(i+m, j+n) \right\} \\ & \ge \sum_{k=NTB}^{K-1} 2^{K-NTB} \cdot \sum_{i,j=0}^{N-1} \left| g_k^t(i,j) - g_k^{t-1}(i+m, j+n) \right| \\ & \ge \sum_{k=NTB}^{K-1} 2^{K-NTB} \cdot \left| \sum_{i,j=0}^{N-1} g_k^t(i,j) - \sum_{i,j=0}^{N-1} g_k^{t-1}(i+m, j+n) \right| . \end{aligned} \quad (10)$$

The last part of (8) can be the lower bound of the TGCBPM matching error in order to apply the SEA method with the results from the previous works. Of course, a lot of candidates point are going to be skipped without performing the block matching process with the result of (8). However, the process to obtain the lower bound is not efficient, because several sum-norm must be calculated depending on the user parameter NTB. Assume that NTB is set as 5, as recommended, then the process calculating the sum norms requires six times only for one candidate search point. Note that the number of sum norms to be calculated depends on the value of NTB.

Note that the gray-code is a well-known code whose mapping function can be easily obtained without bit-transforming, once the length of the code is determined. In order to utilize the fact mentioned above, we define a parameter G^t as follow:

$$G^t(i,j) \equiv \sum_{k=NTB}^{K-1} 2^{K-NTB} g_k^t(i,j) \ . \quad (11)$$

, which is the integer value of gray-code whose length is determined by the given value of the parameters NTB, K and the input pixel value, so the calculation of G^t can be implemented by look-up table.

Applying (1) and (11) into (9), we obtain the main result of the proposed algorithm as follow:

$$CM_{TGC}(m,n) \ge \left| \sum_{i,j=0}^{N-1} G^t(i,j) - \sum_{i,j=0}^{N-1} G^t(i+m, j+n) \right| . \quad (12)$$

Once the value of G^t is obtained in (12), we have only two sum norms left and the number of sum norms does not depend on the value of NTB. In the search process, the right part of (12) is compared with the $CM_{\min}$ which is the up-to-date minimum CM_{TGC} in the search process. We calculate the CM_{TGC} of (9) only if the condition of (12) is satisfied. Otherwise, we skip this search position, move on to the next search position, and save the unnecessary operations. Note that the calculation of sum norms can be done efficiently using the method in [5].

Table I Average PSNR performance (dB) comparison for the algorithms

Sequences	1BT	C1BT	2BT	TGCBPM	Proposed
Akiyo	41.69	42.05	41.88	41.90	41.90
Bus	24.51	25.04	24.91	25.34	25.34
Container	37.29	37.98	37.78	37.49	37.49
Dancer	27.92	28.55	28.84	29.95	29.95
Football	22.89	23.20	23.16	23.77	23.77
Foreman	29.80	30.35	30.09	30.75	30.75
Hall	32.85	34.30	33.64	34.12	34.12
News	31.71	33.65	33.54	34.42	34.42
Stefan	22.92	23.09	23.22	23.50	23.50
Table	29.85	30.53	30.24	30.74	30.74
Average	30.14	30.88	30.73	31.20	31.20

Table II Average reduction ratio of the calculated points (%)

Akiyo	96.13
Bus	54.08
Container	81.01
Dancer	86.02
Football	50.49
Foreman	73.69
Hall	86.94
News	91.37
Stefan	54.01
Table	68.66
Average	74.24

IV. Experimental Results

The propose algorithm is simulated with various video sequences: Akiyo, Bus, Container, Dancer, Football, Foreman, Hall, News, Stefan, and Table. All implemented block matching algorithms are programmed by Visual C++. The motion block size is 16 x 16 pixels and the search range is 33 x 33. The format of video sequence that we used is CIF (352 x 288) and only forward prediction is used. All the searching processes are performed in spiral order. The parameter NTB is set as 5, as recommended in the reference paper.

Table I shows the average PSNR (dB) of the BPM-based ME algorithms and the proposed algorithm. As shown in Table I, the PSNR of TGCBPM based ME is better than that of the other BPM-based ME algorithms. In terms of the PSNR point of view, we can conclude the PSNR performance of the proposed algorithm is totally same as that of the TGCBPM-based ME algorithm from the data of the results in Table I and the mathematical derivation in section III.

In order to campare the performance of the proposed algorithm with that of TGCBPM based ME, we calculate the average reduction ratio of the calculated points. The performance gains of the proposed algorithm somewhat defferent depending on the squences. The reduction ratio of the sequences that have static motion, such as Akiyo and Container, is greater than that of the sequences that have complex motion, such as Football and Stefan. On the average, the performance gain of the proposed algorithm is 74.24% over FS-TGCBPM based ME in terms of the complexity point of view. Note that the average PSNR performance of the proposed algorithm and the TGCBPM based ME is the same.

V. Conclusions

In this paper, a SEA for TGCBPM based ME has been proposed to improve the performance of the TGCBPM based ME in terms of the computational complexity. The proposed algorithm suggests the efficient way to calculate the lower bound for TGCBPM matching criterion and to eliminate the impossible candidates earlier so that the computational complexity of the TGCBPM is reduced significantly. Experimnetal results show that the computational complexity has been reduced substantially, while the PSNR of the proposed algorithm is the same as that of the TGCBPM based ME.

References

[1] R. Li, B. Zeng, and M. Liou, “A new three-step search algorithm for block motion estimation,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 4, pp 438-442, Aug. 1994.

[2] S. Zhu and K. Ma, “A new diamond search algorithm for fast block matching motion estimation,” in *Proc. ICICS*, Singapore, vol. 1, pp 292-296, 1997.

[3] C. Cheung and L. Po, “Normalized partial distortion search algorithm for block motion estimation,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, no. 3, pp 417-422, Apr. 2000.

[4] C. Cheung and L. Po, “Adjustable partial distortion search algorithm for block motion estimation,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 1, pp 100-110, Jan. 2003.

[5] W. Li and E. Salari, “Successive elimination algorithm for motion estimation,” *IEEE Trans. Image Process*, vol. 4, no. 1, Jan. 1995.

[6] B. Natarajan, V. Bhaskaran, and K. Konstantinides, “Low complexity block-based motion estimation via one-bit transforms,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 4, Aug. 1997.

[7] Sarp Ertürk, “Multiplication-free one-bit transform for low-complexity block-based motion estimation,” *IEEE Trans. Signal Processing Letter*, vol. 14, no. 2, Feb. 2007.

[8] Alp Ertürk and Sarp Ertürk, “Two-bit transform for binary block motion estimation,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 7, pp 938-946, Jul. 2005.

[9] Oguzhan Urhan and Sarp Ertürk, “Constrained one-bit transform for low complexity block motion estimation,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 4, pp 478-482, Apr. 2007.

[10] Anıl Çelebi, Orhan Akbulut, Oğuzhan Urhan, and Sarp Ertürk, “Truncated gray-coded bit-plane matching based motion estimation and its hardwqre architecture,” *IEEE Trans. Consum. Electron.*, vol. 55, no. 3, pp 1530-1536, Aug. 2009.

[11] H. Lee and J. Jeong, "Early termination scheme for binary block motion estimation," *IEEE Trans. Consum. Electron.*, vol. 53, no. 4, pp 1682-1686, Nov. 2007.

[12] H. Lee, S. Jin, and J. Jeong, "Early termination scheme for 2BT block motion estimation," *Electron. Lett.*, vol. 45, no. 8, pp 403-405, Apr. 2009.

[13] I. Kim, J. Kim, G. Jeon, and J. Jeong, "Low-complexity block-based motion estimation algorithm using adaptive search range adjustment," *Opt. Eng.*, vol. 51, no. 6, 067010, Jun. 2012.

[14] Y. Wang and G. Tu, "Successive elimination algorithm for binary block matching motion estimation," *Electron Lett.*, vol. 36, no. 24, pp 2007-2008, Nov. 2000.

[15] C. Choi and J. Jeong, "Successive elimination algorithm for two-bit transform motion estimation," *IEICE ELEX*, vol. 7, no. 10, pp. 684-690, May 2010.

[16] C. Choi and J. Jeong, "Successive elimination algorithm for constrained one-bit transform based motion estimation using the Bonferroni inequality," *IEEE Proc. Lett.*, vol. 21, no. 10, pp. 1260-1264, Oct. 2014.