Binary Block Motion Estimation Using an Adaptive Search Range Adjustment Technique

Ilseung Kim and Jechang Jeong
Department of Electronics and Computer Engineering / Hanyang University, Seoul, Korea
Email: ghanjang@gmail.com, jjeong@ hanyang.ac.kr

***Abstract*— This study proposes an adaptive search range adjustment algorithm to enhance the effectiveness of current binary block motion estimation methods. This is because the binary block ME approaches, which perform FS by using different criterion from the generally used sum of absolute differences (SAD), has a significant drawback in PSNR. The combination of the proposed algorithm with the binary block motion estimation is thought to enhance 1) motion estimation accuracy and 2) time efficiency. The findings showed that the proposed combination, compared to the conventional binary block motion estimation method, significantly increased object visual qualities and time efficiency.**

***Index Terms*— Video coding, Motion estimation, Matching criteria, Search range adjustment (SRA)**

I. Introduction

Video compression is crucial to efficiently storing and transmitting digital video data. Currently, block based Motion estimation (ME) and motion compensation (MC) have been regarded as the key techniques of the video compression process because they reduce video data effectively. Although the sum of absolute differences (FS-SAD) employing a full search method is the most frequently used method in the field, the measure also has disadvantages due to the excessive computational loads it requires.

Scholars have therefore proposed a variety of algorithms to reduce the computational burden of the FS-SAD method, including binary block motion estimation techniques. The binary block motion estimation approaches utilize different matching criteria from SAD or SSD to increase the speed of computation of the matching criteria by using binary operations. Using binary operations makes hardware implementation much easier and therefore allows for faster matching criteria computation and a reduction in memory bandwidth. The approaches include one-bit transform (1BT), multiplication-free one-bit transform (MF-1BT), two-bit transform (2BT), constrained one-bit transform (C1BT), and recently proposed truncated gray-coded bit plane matching (TGCBPM), and so on [1]- [5]. Generally, this line of approach emphasizes two key points; how to make a bit-plane(s) to use binary operations and which matching criteria can be used to measure in place of SAD.

The one-bit transform (1BT) method transforms the video frames into a bit-plane by comparing the multi-bandpass filtered version with the original version. The method employs a 17x17 kernel for the comparison and then adopts a matching criterion - the number of non-matching points of 1BT ($NNMP_{1BT}$) shown by

$$NNMP_{1BT}(m,n) = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1}\left\{B^{t}(i,j) \oplus B^{t-1}(i+m, j+n)\right\} \quad (1)$$

where $B^{t}(i,j)$ and $B^{t-1}(i,j)$ are one-bit representations of the current and the reference frame and $\oplus$ denotes the Boolean exclusive-or operation [1].

The multiplication-free one-bit transform (MF-1BT) algorithm employs a multiplication-free 19x19 kernel when creating a bit-plane [2]. In order to enhance the ME accuracy of the 1BT method [3], [4], the 2BT adopts a two-bit transform (2BT) and a constrained one-bit transform (C1BT). 2BT produces two bit-planes by using the local mean, the variance, and the approximated standard deviation. The C1BT reduces the PSNR loss by adding a constrained bit plane. These binary block motion estimation approaches use different matching criteria from SAD but still utilize FS. Thus, several algorithms were proposed to find ways to reduce the complexity.

Wang. *et al*. and Choi. *et al*. adopted an algorithm from a successive elimination algorithm (SEA) based on 1BT and 2BT [6] - [8]. However, the method did not take advantages of the binary block motion estimation: fast computation of the matching criterion and reduced memory bandwidth. Therefore, algorithms which need a significant memory load are not suitable for this method. Urhan *et al.* adopted PDS based on C1BT, and Lee. *et al.* embraced the early termination method based on MF1BT and 2BT [9]-[11]. Low bit-depth matching methods based on ME approaches, however, have a critical drawback because it significantly lowered visual quality in comparison to other fast algorithms. Therefore, using other fast algorithms such as early termination methods with low bit-depth matching approaches is not an ideal option due to the loss in accuracy.

This paper proposes a combination of binary block motion estimation and an adaptive search range adjustment technique as a way to improve visual quality based on the rough distortion measure of binary matching, and a way to reduce computational complexity. The proposed algorithm is also expected to exhibit good

Manuscript received October 9, 2013; revised December 12, 2013.

doi: 10.12720/joace.2.4.376-380

performance, with a wide range of binary matching techniques.

The proposed algorithm will be explained in detail in Section II. Section III will discuss the experiment results by comparing the effectiveness between previously-used algorithms and the proposed algorithm. Finally, conclusions are drawn in Section IV.

II. PROPOSED ALGORITHM

A. Observation

Motion can be divided into two factors; the movement of the camera and the movement of real objects. Camera motion is closely related to the background motion vector, which accounts for the largest part of motion vectors. The background motion vector is neither fast nor complex. Although it is difficult to predict the MV of each block with accuracy when various motions overlap, it is still possible to estimate the tendency, especially, when the tendency of the motion is similar with the surroundings. Thus, this study attempts to demonstrate that the search range can be adaptively determined by separating the motions of the camera from the object. In other words, the focal point of this paper is to control the search range adaptively by determining to which area the current block belongs.

B. Motion Analysis

Statistical modeling was performed, according to the properties of the motion vectors. By doing so, we calculated 5 parameters indicating the statistical features such as the mean and approximated standard deviation in both the frame and the local region.

The following process highly depends on the MVs of the reference frame. It is neither appropriate nor possible to predict the MV of the frame itself when there are significant differences between the MVs of the reference frame and those of the current frame. For that reason, we only performed the search range adjustment algorithm when the the MVs of the reference frame and those of the current frame were similar to each other. The inter-frame reliability term R was used to determine whether the information from the reference frame was in tandem with the current frame. We used the MV of neighboring blocks in the current frame and those of the co-located block in the reference frame to calculate the inter-frame reliability term R, which is shown by:

$$\begin{aligned} R = &\left|MV^{t}_{x-1,y} - MV^{t-1}_{x-1,y}\right| \\ &+\left|MV^{t}_{x-1,y-1} - MV^{t-1}_{x-1,y-1}\right| \\ &+\left|MV^{t}_{x,y-1} - MV^{t-1}_{x,y-1}\right| \end{aligned} \tag{2}$$

where $MV^{t}_{x,y}$ and $MV^{t-1}_{x,y}$ are the motion vector of the current block and that of the co-located block in the reference frame and x, y denote indices of blocks.

After obtaining the inter-frame reliability of term R, we get the global motion vector MV_G and the approximated global standard deviation $\hat{\sigma}_G$ in the frame level. As mentioned in the observation section, the MVs related to the motion of the camera are the largest proportion of the MVs of the whole frame. To reflect this, we define MV_G as the motion vector which is related to the movement of the camera, which is given by:

$$MV_G = \arg\max_{MV}(h(MV)) \tag{3}$$

where h(x) is the histogram of the variable x. Fig. 1 shows the histogram of MV in a frame level. In order to estimate the variation of the MVs in the frame compared to MV_G, we define the approximated standard deviation ($\hat{\sigma}_G$) of the MVs with MV_G, which is given by:

$$\hat{\sigma}_G = (\sum_{x,y}^{N}\left|MV^{t-1}_{x,y} - MV_G\right|)/N \tag{4}$$

where N denotes the number of the motion vectors in the reference frame.

After obtaining the statistical information in the frame level, we need to obtain the statistical information in the local level. In order to measure the variation of the local motions compare to MV_G, we define $\hat{\sigma}_{L1}$ as:

$$\hat{\sigma}_{L1} = (\sum_{i=-1}^{1}\sum_{j=-1}^{1}\left|MV^{t-1}_{x+i,y+j} - MV_G\right|)/9 \tag{5}$$

Finally, we want to find the variation of the motion in the local region itself. To do so, we find the mean (μ_L), which is given by:

$$\mu_L = (\sum_{i=-1}^{1}\sum_{j=-1}^{1} MV^{t-1}_{x+i,y+j})/9 \tag{6}$$

Now, we can estimate the value of the variation in the local region, which is defined as:

$$\hat{\sigma}_{L2} = (\sum_{i=-1}^{1}\sum_{j=-1}^{1}\left|MV^{t-1}_{x+i,y+j} - \mu_L\right|)/9 \tag{7}$$

The notations and the process of this step are shown in Fig. 2.

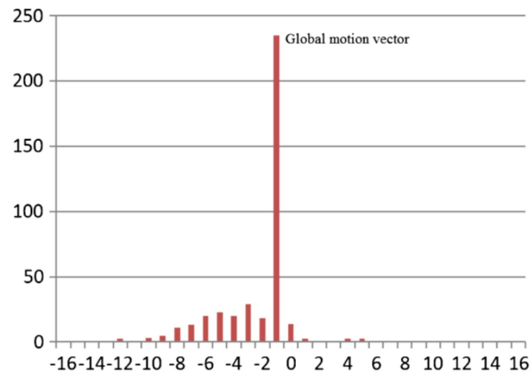

Figure 1. A histogram of motion vectors in Frame #4 of the Foreman sequence

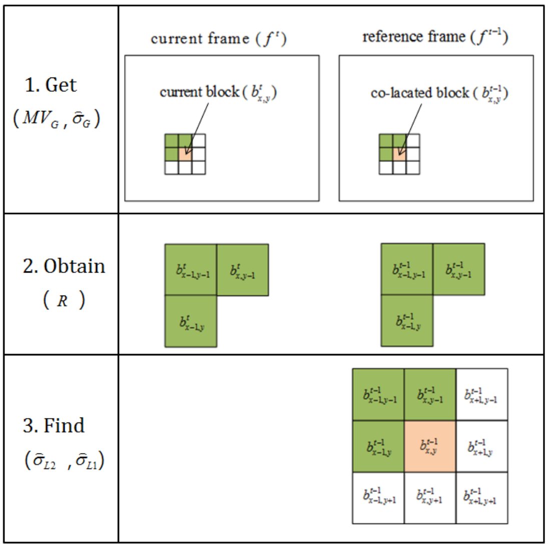

Figure 2. Notations and process of the motion analysis.

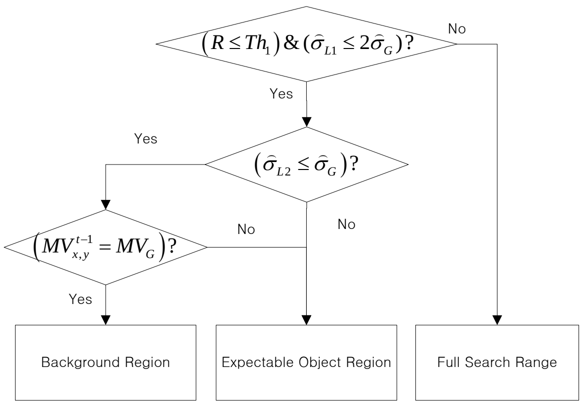

Figure 3. Flowchart for region classification.

C. Region Classification

In order to adjust the search range adaptively, we categorized the blocks into two regions: an expectable background region and an expectable object region. If the block could not be classified into the two regions, we conducted a full search.

First, we verified the two kinds of reliability: the inter-frame reliability and the reliability between the frame and the local region. In order to check the inter-frame reliability, we obtained the term R and the inter-frame reliability is regarded as satisfied when the value of R is less than Th_1. We estimated the complexity of the local region by comparing $\hat{\sigma}_{L1}$ and $\hat{\sigma}_G$. When the local region was too complex compared to the whole frame, we stopped estimating the search range and conducted a full-search.

After checking the reliability, we completed the region classification. If the motion in the local region was static, then there was no need for the wide search range; especially, if the motion was related to the movement of the camera. By comparing $\hat{\sigma}_{L2}$ and $\hat{\sigma}_G$, we estimated the complexity of the local region. If the region was relatively complex, we regarded the region as the expectable object region. It is important to note that we excluded the complex region in the reliability verification stage because it was not predictable. Given that MV_G reflects the movement of the camera, we regarded the block as the background region when the value of $MV^{t-1}_{x,y}$ was same as MV_G. Fig. 3 shows the flowchart of and the conditions used in this step.

D. Determination of the New Search Range

Although the movement velocity is likely to be fixed where the current block is confirmed as the background, we included TH2 and set dynamic search range D in order to find the MV with better accuracy. If the velocity of the camera itself is too fast, it is possible that the search range has deviated from the full search range. Therefore, we clip D from 0 to the full search range. D in the expectable background region is then determined as

$$D = CLIP(MV_G + TH_2, 0, FSR) \tag{8}$$

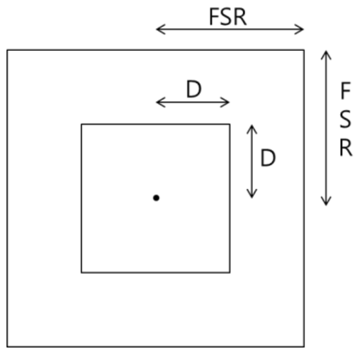

Figure 4. Notations of search range: dynamic search range (D) and full search range (FSR)

where $CLIP(x, p, q)$ clips x to a value between p and q, and FSR is the full search range. TH2 is experimentally set as 2.

In the case of the object, the surrounding object needs to be checked and we need to secure a wider search range in case there is a fast-moving object. Also, it is highly possible that various objects exist in the area where it has been predicted as a single object. To address this issue, we investigated nine blocks including the current block to find the fastest block and reflect it in the search range:

$$\Psi = MAX(MV^{t-1}_{x+i,y+j}), \qquad (i, j \in [-1,1]) \tag{9}$$

where Ψ is the largest value from the absolute value of the MV of the surrounding block. The dynamic search range generally reflects the velocity of the fastest object from the current block and the surrounding block. Considering the complexity of this scenario, and that it cannot be easily predicted, the search needs to be performed by using a wider range in contrast to the case above. Therefore, the search range is set as:

$$D = CLIP(\Psi + TH_2, 0, FSR) \tag{10}$$

Fig. 4 shows the notations of search range; dynamic search range (D) and full search range (FSR).

E. Combination with the Binary Block Motion Estimation

A variety of fast motion estimation algorithms have been studied and one of the major examples includes the binary block motion estimation approaches. However, the approach has significant drawbacks in that it can degrade the qualities. Previous experiments showed that there was an approximate 0.81 PSNR loss for 1BT, 0.57 for 2BT, and 0.43 for C1BT on average. Among the fast algorithms available today, this is the most significant image degradation. Current binary block motion estimation approaches utilize FS and other fast algorithms that can be relatively easily developed. However, such fast algorithms inevitably lead to additional PSNR loss. The combination of the binary block motion estimation methods and the proposed scheme can address both issues. By controlling the search range adaptively by particular condition, we can maintain and or enhance visual quality by avoiding the wrong MV and we can drastically reduce time expenditure.

III. Experimental Results

In order to compare the proposed algorithm to the conventional algorithms, we simulated various video sequences–Akiyo, Bus, Coastguard, Flower, Hall-monitor, Mobile, Paris, and Stefan. To draw a fair comparison, the conventional algorithms are implemented first, followed by the proposed algorithm by adding SRA under the exact same conditions. All BMAs that are implemented were programmed using Visual C++. MB size is 16 x 16 pixels and the search window was 33 x 33 pixels. The image format we used was CIF (352 x 288), and only forward prediction was used.

The results showed that the proposed SRA lost only about a 0.01 dB PSNR and it was approximately 4.24 times faster than FS. In terms of time reduction, sequences with static motion (e.g., Akiyo) were much faster than sequences with a fast and complex motion (e.g., Bus, Stefan), but this result was unsatisfactory.

TABLE I. PSNR ($\triangle$ dB) Comparison

	FS	*1BT [1]*	*2BT [3]*	*C1BT [4]*
Akiyo	0.00	0.14	0.04	0.00
Bus	-0.03	0.06	0.03	0.02
Coastguard	0.00	0.03	0.04	0.02
Flower	-0.01	0.01	0.04	0.00
Hall-monitor	-0.02	0.58	0.14	0.04
Mobile	-0.02	0.02	0.01	-0.01
Paris	0.00	0.02	0.05	-0.01
Stefan	-0.01	0.01	0.04	0.00
Average	-0.01	0.11	0.05	0.01

As mentioned earlier, the binary block motion estimation methods degrade the image quality the most among all fast algorithms that have been researched thus far. However, the proposed SRA method compensates the matching errors originated from binary matching. As a result, the findings showed an approximately 0.11dB PSNR gain for 1BT, 0.05dB for 2BT and 0.03dB for C1BT on average as shown in Table I. In terms of time reduction, the proposed algorithm was approximately 2.85 times faster for 1BT, 2.90 times faster for 2BT, and 4.25 times faster for TGCBM, compared to the conventional algorithms shown in Table II.

TABLE II. Speed-up Ratios (Times) Compared to Conventional Algorithms.

	FS	*1BT [1]*	*2BT [3]*	*C1BT [4]*
Akiyo	8.71	7.62	8.48	10.66
Bus	2.30	1.16	1.08	1.05
Coastguard	3.95	1.34	1.12	1.25
Flower	3.14	2.30	2.38	3.41
Hall-monitor	4.40	2.05	1.56	10.32
Mobile	4.68	2.69	2.86	2.45
Paris	4.86	4.35	4.51	3.67
Stefan	1.89	1.26	1.23	1.18
Average	4.24	2.85	2.90	4.25

IV. Conclusion

This study proposed to combine the adaptive search range adjustment algorithm, which can be predicted by using MV information from the reference frame, with the binary block motion estimation approaches. The binary matching method has gained support as a fast algorithm when employing SAD as matching criterion. There are, however, still problems remaining with these approaches including significant PSNR loss and the requirements of a full search. Finding ways to effectively address such issues was the main purpose of this study.

By analyzing the motion information, we divided the image into two groups and applied adaptive search range techniques for each group. To do so, the inter-frame reliability and the complexity of neighbouring objects compared to that of the reference frame were considered. By combining this method with the previous binary block motion estimation methods, we were able to obtain an increase of 0.05dB in PSNR and a speed of 3.3 times faster.

Acknowledgments

This research was supported by the MSIP (Ministry of Science, ICT & Future Planning), Korea, under the ITRC (Information Technology Research Center) support program supervised by the NIPA (National IT Industry Promotion Agency)" (NIPA-2013-H0301-13-1011)

References

[1] 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, no. 4, pp. 702-706, Aug 1997.

[2] S. Ertürk, “Multiplication-free one-bit transform for low-complexity block-based motion estimation,” *IEEE Trans. Circuit Syst. Video Technol.*, vol. 14, no. 2, pp. 109- 112, Feb 2007.

[3] A. Ertürk and S. Ertürk, “Two-bit transform for binary block motion estimation,” *IEEE Trans. Circuit Syst. Video Technol.*, vol. 15, no. 7, pp. 938- 946, July 2005.

[4] O. Urhan and S. Ertürk, "Constrained one-bit transform for low complexity block motion estimation," *IEEE Trans. Circuits and Systems for Video Tech.*, vol. 17, no. 4, pp. 478 – 482, 2007.
[5] A. Celebi, O. Akbulut, O. Urhan, and S. Ertürk, "Truncated gray-coded bit-plane matching based motion estimation and its hardware architecture," *IEEE Trans. Circuits and Systems for Video Tech.*, vol. 55, no. 3, pp. 1530–1536, 2009.
[6] Y. Wang and G. Tu, "Successive elimination algorithm for binary block matching motion estimation," *IEE Electron. Lett.*, vol. 36, no. 24, pp. 2007-2008, Nov 2000.
[7] C. Choi and J. Jeong, "Successive elimination algorithm for two-bit transform-based motion estimation," *IEICE Electronics Express*, vol. 7, no. 10, pp. 684-690, 2010.
[8] W. Li and E. Salari, "Successive elimination algorithm for motion estimation," *IEEE Trans. Image Processing*, vol. 4, pp. 105–107, Jan 1995.
[9] O. Urhan, "Constrained one-bit transform based motion estimation using predictive hexagonal pattern," *Journal of Electron. Imaging*, vol. 61, no. 3, July-Sep 2007.
[10] H. Lee and J. Jeong, "Early termination scheme for binary block motion estimation," *IEEE Trans. Consumer Electron*. vol. 53, no. 4, pp. 1682-1686, Nov 2007.
[11] H. Lee, S. Jin, and J. Jeong, "Early termination algorithm for 2BT block motion estimation," *IEE Electron. Lett.*, vol. 45, no. 8, pp. 403-405, April 2009.

Ilseung Kim received a B.S. degree in electronic, electrical and computer engineering from Hanyang University, Korea, in 2010. He is currently a candidate for a Ph.D. degree in electrical, computer and communication engineering at Hanyang University, respectably. His research interests include video coding and image processing.

Jechang Jeong received a B.S. degree in electronic engineering from Seoul National University, Korea, in 1980, a M.S. degree in electrical engineering from the Korea Advanced Institute of Science and Technology in 1982, and a Ph.D. degree in electrical engineering from the University of Michigan, Ann Arbor, in 1990. From 1982 to 1986, he worked with the Korean Broadcasting System, where he helped develop teletext systems. From 1990 to 1991, he worked as a postdoctoral research associate at the University of Michigan, Ann Arbor, where he helped develop various signal-processing algorithms. From 1991 through 1995, he worked for Samsung Electronics Company, Korea, where he was involved in the development of HDTV, digital broadcasting receivers, and other multimedia systems. Since 1995, he has conducted research at Hanyang University, Seoul, Korea. His research interests include digital signal processing, digital communication, and image and audio compression for HDTV and multimedia applications. He has published numerous technical papers. He received the Scientist of the Month award in 1998, from the Ministry of Science and Technology of Korea, and was the recipient of the 2007 IEEE Chester Sall Award and 2008 ETRI Journal Paper Award. He was also honored with a government commendation in 1998 from the Ministry of Information.