Fast Motion Estimation by Adaptive Early Termination

Esam A. Al_Qaralleh
Electronic engineering, NCTU
HsinChu, Taiwan
esam@twins.ee.nctu.edu.tw

Tian-Sheuan Chang, *Member, IEEE*
Electronic engineering, NCTU
HsinChu, Taiwan
tschang@twins.ee.nctu.edu.tw

***Abstract*—** This paper presents a fast motion estimation (ME) algorithm by adaptively changing the early termination threshold for the current accumulated partial SAD value. The simulation results show that the proposed algorithm can provide the similar quality while save 77.9% and 50.6% of SAD computation when comparing with that in MPEG-4 VM18.0 and H.264 JM9.0, respectively.

I. INTRODUCTION

Motion estimation (ME) has been adopted by all of the existing international standards related to video coding [1]-[3] to remove temporal redundancy between frames. However, ME is also the most computationally intensive part in a typical video encoder (50% - %90) of the entire system [4]. Thus, many fast ME algorithms have been proposed to reduce the complexity by searching only a subset of eligible candidates, e.g. three step search algorithm, global elimination algorithm (GEA) [4] and quarter-pel motion estimation (QME) [5]. GEA, derived from successive elimination algorithm (SEA) [6], is proposed to remove the problems of SEA by using techniques similar to pel averaging and multiple candidates search. QME sub-samples the search window by four to speed up the process.

Other fast algorithms employ early termination scheme as in Hilbert-grouped partial distortion search (HGPDS) [7] and adjustable partial distortion search (APDS)[8]. HGPDS groups the pixels for computing the partial distortion according to their activities in a Hilbert scan. APDS divides each matching block into 16 equal sized group, and starts matching by accumulating one group after another to the matching process.

One early termination scheme, partial distortion search algorithm (PDS), is recommended to be used in MPEG-4 and H.264 reference software [3][9] to reduce the computational complexity of the SAD (sum of absolute difference) without introducing any loss in PSNR quality. In the PDS, the accumulated partial SAD (PSAD) is used to eliminate the impossible candidates of motion vector before the completion of calculating the SAD in a matching block. Thus, if the PSAD is greater than or equal to the current minimum SAD at anytime, this candidate is rejected and the remaining SAD computation is skipped. The PSAD is computed and accumulated by calculating the SAD for one line of the block at a time. Though this scheme can skip some unnecessary SAD computations, there is still room to be improved. For example, this scheme cannot efficiently skip the SAD computations when the candidate SAD is similar to the current minimum SAD. This situation will happen especially at the search points near to the predicted or the final motion vector location.

To solve above problems, we propose an early termination algorithm by adaptive threshold instead of nearly constant minimum SAD value during the SAD accumulation. The result shows that it can reduce the SAD computation significantly with negligible quality degradation.

The rest of the paper is organized as follows. In Section II, we will introduce the proposed early termination scheme. Section III will show the experimental results. Section IV will present comparison with other fast ME algorithms. Finally the conclusion is made on Section V.

II. EARLY TERIMINATION ALGORITHM

Fig. 1(a) shows the flow of the PDS. It starts by assuming a minimum SAD (SAD_{min}), by computing the SAD_{pred} at the predicted location in the search window, and uses that value as current minimum SAD for later comparison. Then it starts SAD computation for each search point. During the SAD computation, if the partial SAD is equal to or greater than the minimum SAD, it terminates the remaining partial SAD computation and jumps to the next search point. For such early termination purpose, the partial SAD is computed one line at a time and accumulated to the total SAD for that MB. This method can quickly skip the unnecessary SAD computation and preserve the search quality. However, if the SAD of the candidate MB is similar to the current minimum SAD, there is little room to skip the computations.

Fig. 1(b) shows the flow of the proposed algorithm. Our proposed algorithm adopts the similar approach as the PDS. We still compute the partial SAD one line at a time and add it to the total SAD. However, during the accumulation of SAD, we use a threshold value SAD_{TH} instead of the minimum SAD as an early termination condition. If the accumulated SAD value is larger than the threshold, we skip

the remaining SAD computation and proceed to the next search point. Otherwise, we continue the SAD computation and accumulation, and update the SAD_{TH} accordingly. When a match occurs, which means lower SAD value than the current minimum SAD, SAD_{TH} is modified according to the new value of minimum SAD. To help adaptively change the threshold value, three terms are introduced in the flow diagram, "TH", "error", and "dec". "TH" represents the base threshold value used in early termination process. The term "error" indicates the space we allow for each line to pass the matching condition, if the PSAD for that line is slightly greater than SAD_{TH}. This "error" is reduced for every time we process a new line by subtracting "dec" from SAD_{TH}. Thus at the end of the matching process, SAD_{TH} would be equal to min_SAD. The values for TH, error and dec for H.264 and MPEG-4 are depicted in Fig. 2.

The advantage of our algorithm is that with the adaptive threshold value instead of the constant one allows us to skip more unnecessary SAD computation.

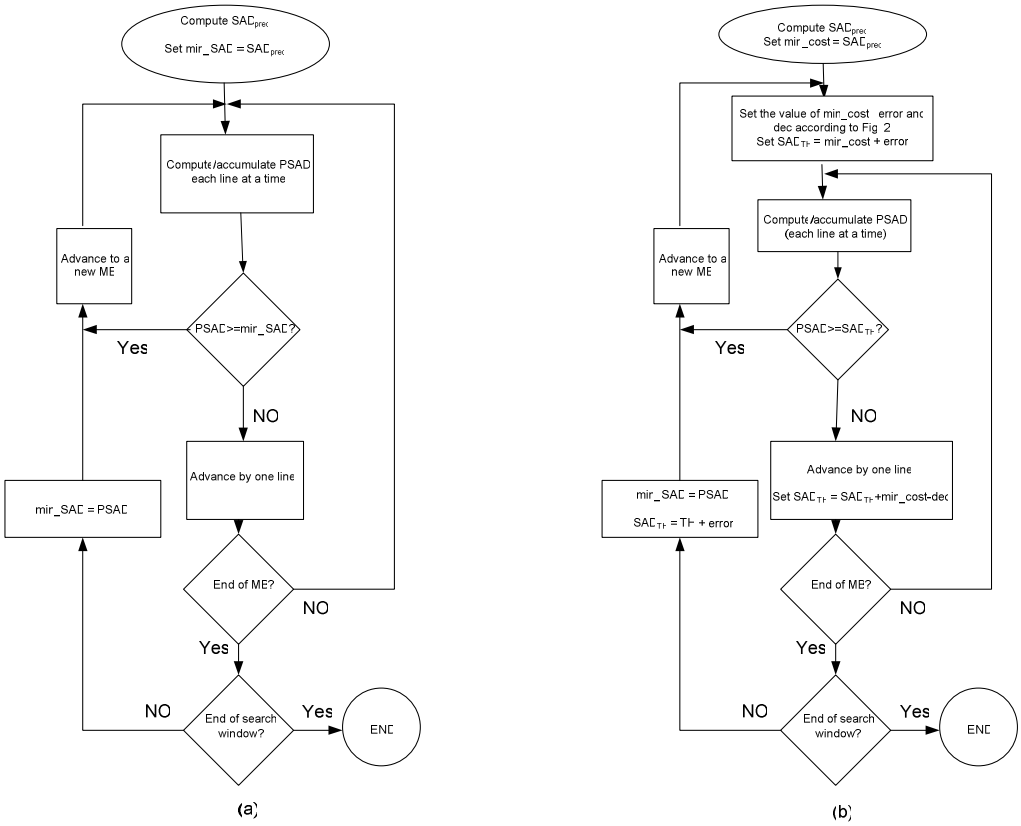

Figure 1. (a) algorithm flow for the PDS, and (b) our proposed algorithm.

III. EXPERIMENTAL RESULTS

In the following, we will show the simulation results of the proposed algorithm by integrating it into the MPEG-4 verification model V18.0 [3], and H.264 JM 9.0 [9]. All the following test sequences are in CIF format of 300 frames with ±16 search range, unless specified.

Like other early termination algorithms, the efficiency of the proposed algorithm depends on both, the scan pattern to the macroblock (MB), and the first SAD_{TH} value that affects how fast we can reach the minimum SAD point and save more computations. In the following test, we test two scan patterns as depicted in Fig. 3, the spiral scan adopted by MPEG-4 VM18.0 and the normal raster scan used for most of the hardware implementations. SAD_{TH} should be set to a value that ensures fast early termination and gives accurate results as well. Choosing SAD_{TH} too large, the termination process will be slower, especially for the full scan method. In the following testing results, the first SAD_{TH} is set to be SAD_{pred}, which is the SAD calculated at the predicated motion vector position.

Table I shows the results for both scanning methods, relative to the full search algorithm without any early termination. The comparison results are produced and tabulated according to the parameters as below:

CHG_BIT: Change of bits used for the whole sequence.

CHG_COMPLEXITY: Change of SAD computations for the whole sequence.

CHG_PSNR: Change of PSNR.

H.264

if(mode== 1) {mir_cost=mir_cost/16 error = 64 dec=4;}
if(mode== 2) {mir_cost=mir_cost/8 error = 32 dec=4;}
if(mode== 3) {mir_cost=mir_cost/16 error = 32 dec=2;}
if(mode== 4) {mir_cost=mir_cost/8 error = 16 dec =2;}
if(mode== 5) {mir_cost=mir_cost/4 error= 8 dec =2;}
if(mode== 6) {mir_cost=mir_cost/8 error= 8 dec = 1;}
if(mode== 7) {mir_cost=mir_cost/4 error= 4 dec=1;}
SAD_{TH} = mir_cost + error

MPEG-4

mir_cost = mir_SAD/16 error = 64 dec=4
SAD_{TH} = mir_cost + error

Figure 2. Values for TH, error, and dec for different block sizes in H.264 and that for MPEG-4

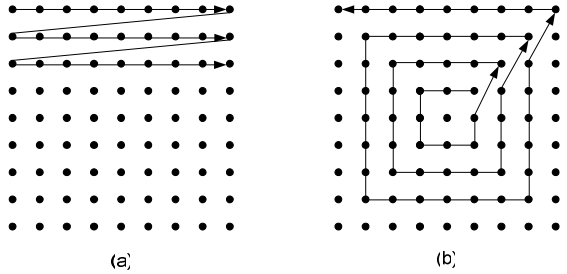

Figure 3. Scan patterns in the search window, (a) raster scan , and (b) spiral scan.

TABLE I. EXPERIMENTAL RESULTS FOR THE PROPOSED METHOD WITH Q_P = 16, AND 4MV OPTION DISABLED

	CHG_BIT (%)		CHG_PSNR		CHG_COMPLEXITY (%)	
Test sequence	*raster scan*	*spiral scan*	*raster scan*	*spiral scan*	*raster scan*	*spiral scan*
Foreman	1.77	1.04	-0.01	-0.01	-75.3	-86.1
Stefan	2.53	2.35	0.00	0.00	-76.2	-87.4
News	1.30	1.02	0.00	0.02	-74.9	-91.9
Coastguard	0.96	0.51	-0.01	-0.01	-78.4	-91.1
Tempet	3.08	-0.08	-0.02	0.00	-77.3	-90.8
M&D	1.28	0.07	-0.04	0.01	-70.8	-86.6
Mobile	2.21	0.90	-0.01	-0.01	-81.8	-91.8
Total/Average	1.88	0.83	-0.01	0.00	-76.39	-89.39

TABLE II. EXPERIMENTAL RESULTS FOR MPEG-4 WITH SPIRAL SCAN PATTERN, Q_P = 16, AND 4MV ENABLED.

	Bit rate			PSNR			Complexity		
Test sequence	*VM 18.0*	*Proposed*	*CHG_BIT (%)*	*VM 18.0*	*Proposed*	*CHG_PSNR*	*VM 18.0*	*Proposed*	*CHG_COMPL-EXITY (%)*
Foreman	1806074	1818842	0.71	31.8644	31.8641	-0.0003	7.19E+08	2.02E+08	-71.919
Stefan	8023412	8157877	1.68	28.4128	28.4085	-0.0043	8.02E+08	2.66E+08	-66.853
News	1109574	1113001	0.31	31.9215	31.9281	0.0066	1.48E+09	1.65E+08	-88.882
Coastguard	3522424	3534403	0.34	29.1339	29.1293	-0.0046	8.37E+08	2.12E+08	-74.716
Tempet	4278248	4256592	-0.51	27.9786	27.9769	-0.0017	5.99E+08	1.61E+08	-73.156
M&D	668805	669789	0.15	34.0386	34.0355	-0.0031	1.38E+09	2.75E+08	-80.062
Mobile	7941210	7970559	0.37	26.3518	26.3463	-0.0055	6.06E+08	1.39E+08	-77.081
Total/Average	27349747	27521063	0.63			-0.0018	6.43E+09	1.42E+09	-77.914

It is clear that the spiral search pattern achieves better results in terms of total bits and PSNR, and also saves more SAD operations than the raster scan method also gives comparable results. This is due to the raster scan pattern will pass by many locale minimum, change SAD_{TH} accordingly and may skip the targeted MB. This effect is shown clearly in the low and moderate motion test sequences, such as Tempet, M&D and Mobile, where the motion vector is more likely to be near to the predicated position.

Table II shows the comparison results with MPEG-4 VM 18.0. The proposed early termination method is also applied to the 8x8 sub-block motion estimation. Our method can save 77.91% of complexity with negligible quality loss.

The same tests are applied to H.264. TABLE III shows the results for the proposed method compared to the early termination method employed in JM 9.0. The proposed method can save 50.6% of complexity with similar quality.

The complexity saving is due to the adaptive threshold mechanism during the SAD accumulation. Every time SAD_{TH} adapted to a lower value, more line skipping will occur. Only those MBs near to the lowest SAD point will consume more SAD computations because the error between those MBs and the current MB will be smaller, and many lines will pass the threshold detection.

IV. COMPARISON WITH OTHER ALGORITHMS

When compared with other early termination algorithms such as [7] (64.5% of complexity saving) and [8] (61.45% of complexity saving), the presented algorithm is superior to others. For other non-early termination algorithms, TABLE IV presents a comparison between our proposed algorithm with recently proposed GEA and QME. It is clear that the proposed method can save more computations with negligible quality loss.

V. CONCLUSION

In this paper we present a simple but efficient early termination method. It reduces the complexity of motion estimation module by skipping the unnecessary SAD computations with adaptively changed threshold. Both MPEG-4 and H.264 can make benefit of this method. The reduction in SAD computations achieved 77.9% and 50.6 % saving for MPEG-4 VM18.0 and H.264 JM9.0 respectively with negligible quality loss.

REFERENCES

[1] ISO AEC JTC1/SC29/WG11, N2502a, Generic coding of Audio-Visual Objects: Visual 14496-2, Final Draft IS, Atlantic City, Dec. 1998.

[2] Joint Video Team (JVT), "Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specifica- tion," 7th Meeting, Pattaya, Thailand, March 7-14, 2003.

[3] ISO/IEC JTC1/SC29/WG11 N3908, "MPEG-4 Video Verification Model Version 18.0," Jan. 2001.

[4] Yu-Wen Huang; Shao-Yi Chien; Bing-Yu Hsieh; Liang-Gee Chen,"An efficient and low power architecture design for motion estimation using global elimination algorithm," in Proc. Acoustics, Speech, and Signal Processing (ICASSP '02), May 2002, vol. 3, pp. 3120-3123.

[5] Kun-Bin Lee; Hao-Yun Chin; Hui-Cheng Hsu; Chein-Wei Jen, "QME: an efficient subsampling-based block matching algorithm for motion estimation," IEEE International Symposium on Circuits and Systems, Vol. 2, Page(s):II - 305-8, May 2004

[6] W. Li and E. Salari, "Successive elimination algorithm for motion estimation," *IEEE Trans. on Image Processing*, vol. 4, no. 1, pp. 105–107, Jan. 1995.

[7] Yui-Lam Chan; Wan-Chi Siu,"An adaptive partial distortion search for block motion estimation," IEEE International Conference on Acoustics, Speech and Signal Processing, Vol.3, pp.153-156, April 2003, Hong Kong.

[8] Chun-Ho Cheung; Lai-Man Po,"Adjustable partial distortion search algorithm for fast block motion estimation," IEEE Trans. on Circuits and Systems for Video Technology, Vol.13, No. 1, pp. 100-110, Jan. 2003.

[9] JVT reference software version 9.0 http://bs.hhi.de/~suehring/tml/download/.

TABLE III. EXPERIMENTAL RESULTS FOR H.264 WITH SPRIRAL SCAN PATTERN, Q_P = 28, ±32 SEARCH RANGE , AND RD-OPTIMIZATION DISABLED.

	Bit rate			PSNR			Complexity		
Test sequence	*JM*	*Proposed*	*CHG_BIT (%)*	*JM*	*Proposed*	*CHG_PSNR*	*JM*	*Proposed*	*CHG_COMPLEXITY (%)*
Foreman	3223016	3240832	0.55	37.01	36.97	-0.04	3.07E+10	1.66E+10	-45.89
Stefan	13694704	13710984	0.12	35.4	35.37	-0.03	5.43E+10	2.74E+10	-49.47
News	2228896	2244720	0.71	38.09	38.07	-0.02	2.83E+10	1.52E+10	-46.35
Coastguard	11279688	11231760	-0.42	34.52	34.51	-0.01	6.78E+10	3.08E+10	-54.49
M&D	1382624	1382808	0.01	38.93	38.92	-0.01	2.86E+10	1.40E+10	-50.92
Mobile	17495736	17458936	-0.21	33.76	33.74	-0.02	5.37E+10	2.60E+10	-51.59
Total/Average	49304664	49270040	-0.07			-0.02	2.63E+11	1.30E+11	-50.60

TABLE IV. COMPARISON FOR DIFFERENT ALGORITHMS

	CHG_BIT			CHG_PSNR			CHG_COMPLEXITY		
Test sequence	*GEA (Δ%) [4]*	*QME (Δ%) [5]*	*Proposed (%)*	*GEA (ΔdB)[4]*	*QME (ΔdB)[5]*	*Proposed (Δ dB)*	*GEA (Δ%) [4]*	*QME (Δ%) [5]*	*Proposed (Δ%)*
Foreman	0.77	0.41	1.20	-0.0222	-0.0059	-0.0046	-66.41	-60.49	-86.1
Stefan	0.85	0.15	1.87	-0.0317	-0.0123	-0.0077	-69.40	-67.41	-87.4
News	0.45	0.34	0.67	-0.0384	-0.0176	-0.0062	-39.09	-66.86	-91.9
Coastguard	0.95	0.09	0.31	-0.0018	-0.0003	-0.0058	-72.92	-65.21	-91.1
M&D	0.67	0.48	0.61	-0.0665	-0.0234	0.0112	-58.99	-39.86	-86.6
Mobile	0.29	0.12	0.48	-0.0027	0	-0.0078	-73.24	-59.36	-91.8