Adaptive Search Range Determination for Fast Motion Estimation

Wonjin Lee
Department of Electronics and Computer Engineering
Hanyang University
17 Haengdang-dong, Seongdong-gu, Seoul, Korea
veronica0083@gmail.com

Jechang Jeong
Department of Electronics and Computer Engineering
Hanyang University
17 Haengdang-dong, Seongdong-gu, Seoul, Korea
jjeong@hanyang.ac.kr

***Abstract*—The motion estimation (ME) in video codec is extremely important, having acritical effect on encoding time and video quality. Although the full search algorithm is the most fundamental ME method which shows the best video quality, it has high computation complexity. To alleviate this issue, many literatures have been proposed to improve the computational speed and maintain the video quality. In this paper, we propose the new method which determines the search range by using the sum of absolute difference between macroblocks of current frame and reference frame. Experimental results show that proposed method can achieve nearly 209 times computation reduction and can maintain its mean square error performance very close to full search method.**

Keywords- Block matching, partial distortion search, video coding, motion estimation, search range determination.

I. INTRODUCTION

Motion estimation (ME) in many video standards uses the block matching algorithm (BMA). It efficiently removes the temporal redundancy between frames [1-3]. BMA divides a frame into macroblocks and searches the most similar prediction block with the macroblock of current frame.

Instead of sending all information of current macroblock, BMA only encodes difference between current macroblock and prediction block and motion vector that indicates the relative location of prediction block. The full search (FS) algorithm has high computation complexity because of comparing the sum of absolute difference (SAD) in all locations within search range to find the most similar block with current macroblock. Therefore, FS occupies the most of whole encoding time, while it gives the smallest distortion.

In order to reduce the computation complexity of FS, many literatures have been proposed.These algorithms can be classified into fast searching approaches and fast matching approaches.The fast searching approaches can achieve the high speed-up by reducing the search point in search window.

The representative methods are diamond search (DS) [4], four-step search (4SS) [5], new three-step search (N3SS) [6].The representative methods to reduce the encoding time using sub samples are partial distortion search (PDS), normalized partial distortion search (NPDS) [7], adjustable partial distortion search (APDS) [8], and efficient two step edge based partial distortion search for fast block motion estimation (TS-EPDS) [9]. Previous algorithms tried to improve the search speed while maintaining the peak signal-to-noise ratio (PSNR) performance.

In this paper, we present the method that determine the search range by using SAD between macro block of current frame and prediction block of reference frame to reduce the computation complexity and maintain the video quality. The rest of this paper is organized as follows. Section 2 introduces the previous idea and preliminaries. Section 3 explains the proposed method and experimental results are exhibited in Section 4. Finally, concluding remarks are given in Section 5.

II. PREVIOUS ALGORITHMS

NPDS separates 16x16 macro block into 4x4 blocks that do not overlap and obtain the partial distortion between current macro block and prediction block. Partial distortion is SAD for one group which consists of a total of 16 pixels extracted to each 4x4 block, as shown in Figure 1.

The SAD for one group is defined as

$$d(p)(k,l;u,v) = \sum_{i=0}^{3}\sum_{j=0}^{3}\left| \begin{array}{l} I_n\left(k+4i+s(p),\, l+4j+t(p)\right) \\ -I_{n-1}\left(k+4i+s(p)+u,\, l+4j+t(p)+v\right) \end{array} \right|, \quad (1)$$

where pixel position of p is given in TABLE 1, I is a frame, k and l are the positions of current macro block, u and v are position of candidate blocks in search range, n and n-1 are the current frame and the previous frame, respectively. The $t(p)$ and $s(p)$ are offset of the location for the p^{th} partial distortion. $d(p)$ is partial distortion that is the SAD for one group and it is accumulated $D(p)$ and $D(p)$ is compared with full SAD of starting search point.

$D(p)$ is the accumulated d(p), which is given by

$$D(p) = \sum_{i=0}^{p} d(i) \quad . \quad (2)$$

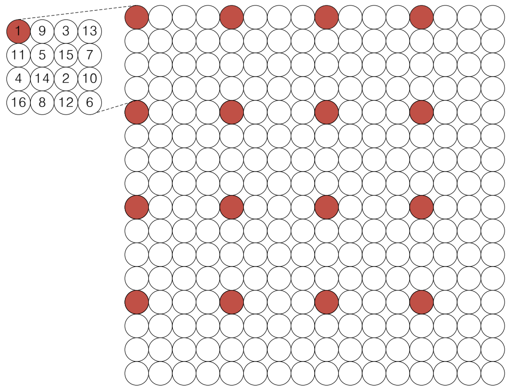

Figure 1. Pixel group for partial distortion.

TABLE I. PIXEL POSITION OF P

p	(s(p), t(p))	p	(s(p), t(p))
1	(0,0)	9	(1,0)
2	(2,2)	10	(3,2)
3	(2,0)	11	(0,1)
4	(0,2)	12	(2,3)
5	(1,1)	13	(3,0)
6	(3,3)	14	(1,2)
7	(3,1)	15	(2,1)
8	(1,3)	16	(0,3)

NPDS is compared with initial SAD that is calculated from all the pixels for 16x16 block, while accumulating the partial SAD for each 16 group. It can improve the match speed by comparing initial SAD with $D(p)$*16/p. The $D(p)$ can estimate initial SAD by multiplying 16/p. For instance, if SAD of each 16 group is similar, we can estimate the full SAD with SAD of one group. However, if SAD of each 16 group is not similar, NPDS can find the motion vector incorrectly. The APDS that is proposed to solve this demerit separates the first group of 16 groups of NPDS into 4 groups to increase search speed. Instead of increasing the search speed by dropping the PSNR, APDS presents the quality factor to increase search speed and maintain the PSNR of FS.

Equation of the quality factor is given by

$$f(n,k) = (1-k)n + kN^2 , \quad (3)$$

where N is the normalization factor and n is the number of accumulated set of pixels. The n is from 1 to 16. *k*has the value from 0 to 1. If *k*=1, APDS has the same performance as PDS, if *k*=0, APDS has the same performance as NPDS.

TABLE II. PROBABILITY OF EACH SEARCH RANGE

SR \ SAD	128	256	512	1000	1500
1	100	97	94	93	92
2	100	98	95	94	93
3	100	98	96	95	95
4	100	99	97	96	96
5	100	99	97	97	97
6	100	99	97	97	97
7	100	99	98	97	97
8	100	99	98	98	98
9	100	99	98	98	98
10	100	99	99	98	98
11	100	100	99	99	99
12	100	100	99	99	99
13	100	100	99	99	99
14	100	100	99	99	99
15	100	100	100	99	99
16	100	100	100	100	100

III. PROPOSED ALGORITHM

APDS algorithm calculates the SAD for the macroblock at the same position in reference frame with the macroblock in current frame. Then, in the next candidate blocks, APDS predicts the SAD of all the pixels by using the small samples and performs while macroblock moves whole search range. If the SAD of the first candidate is small enough, the true motion vector may be near the current position because difference between macroblocks of current and reference is small.

TABLE 2 presents the probability of that there is a true motion vector at each search range (SR) according to the first calculated SAD. we can find that probability of true motion vector is high in small search range when the first calculated SAD is small enough.

From this motivation, we propose a new method that searches the partial range, it does not perform whole search range. However, this method may find inaccurate motion vector by falling into local minimum point.

To solve this demerit, after we calculate the SAD of median of three neighboring motion vectors and the SAD of origin, we set the position of SAD with a smaller value to the starting point (SP). Then we adjust the size of search range according to the initial SAD. The equation of determination of SR according to the initial SAD is defined as

$$SR = ceil(\frac{SAD_{initial}}{TH}) , \quad (4)$$

where we experimentally defined TH by 256 to maintain the approximately 95% in the TABLE 4.

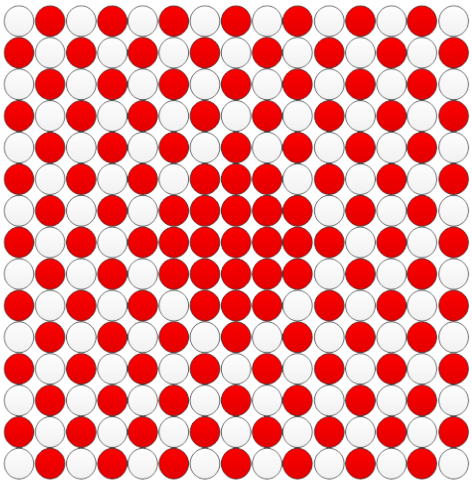

Figure 2. Search point of the proposed method.

We set the search point to increase the search speed as shown in Figure 2. For instance, we find the motion vector roughly and search the neighboring skipped point. Flowchart of proposed method is given by Figure 3.

The Proposed method is performed as follows :

Step 1)
Find the minimum SAD between the SAD of median of three neighboring motion vectors and the SAD of origin.

Step 2)
Determine the search range with the use of SR equation and starting point that is position of minimum SAD decided in Step 1.

Step 3)
Perform APDS through the use of proposed search pattern in Figure 2.

Step 4)
If there are neighboring skipped points, perform APDS in skipped points of neighboring 8 points, otherwise, go to Step 5.

Step 5)
Get the motion vector with the minimum SAD of current block.

IV. EXPERIMENTAL RESULTS

We conducted experiments through the use of 10 sequences (Akiyo, Bridge_close, Children, Hall, Mother, News, Silent, Singer, Stefan, and Paris). The proposed method and BMAs are performed to only motion estimation, not codec such as MPEG-2 and H.264/AVC. Each sequence has 300 frames and the sequence format is CIF(352x288). All implemented BMAs are programmed by visual C++. The proposed method is compared with four conventional methods: FS, PDS, NPDS, APDS, and TS-EPDS. The size of macroblock used in motion estimation is 16x16. We set the default search range by $\pm$16.

The PSNR, speed-up are used to evaluate the objective performances. The speed-up is computed as operations of

Figure 3. Flowchart of the proposed method.

FS divided by operations of BMA. The total number of operations is the sum of addition, comparison, absolution, and multiplication.

As shown in TABLE 3, the proposed method resulted in the average of 0.02 dB PSNR degradations compared to that of FS. The PSNR performance of APDS and proposed method show almost same with FS, but TS-EPDS has significant PSNR drop on News, Singer, and Stefan sequences.

As tabulated in TABLE 4, theaverage of speed up of the proposed method was 209 times faster than FS, 15 times faster than NPDS and 8 times faster than APDS, and 2 times faster than TS-EPDS.

The proposed algorithm maintained the similar level of PSNR with previous fast search algorithm and could check that computational complexity is considerably less than the previous algorithms. Furthermore, if the background of video sequence is stopped, we could confirm that complexity is reduced exceptionally.

V. CONCLUSION

In this paper, we proposed the algorithm changing the search range for fast search. The proposed method can adjust search range considering that if SAD between current and candidate macroblocks is small enough, the true motion vector may be near the current position. As the experimental results suggest, the proposed method reduced the encoding time by maintaining the similar level of PSNR compared to previous fast search methods. If starting point is determined more accurately, our proposed method can be further improved.

TABLE III. COMPARISON OF THE AVERAGE PSNR WITH THE CONVENTIONAL METHODS

	PSNR					
	FS	**PDS**	**NPDS**	**APDS**	**TS-EPDS**	**Proposed**
Akiyo	42.94	42.94	42.85	42.94	42.92	42.94
Bridge_close	35.23	35.23	35.23	35.23	35.21	35.23
Children	29.79	29.79	29.58	29.77	29.77	29.76
Hall	34.83	34.83	34.70	34.80	34.77	34.79
Mother	40.44	40.44	40.35	40.43	40.41	40.41
News	36.90	36.90	36.68	36.88	36.82	36.86
Silent	35.85	35.85	35.69	35.83	35.83	35.84
Singer	36.87	36.87	36.62	36.82	36.76	36.81
Stefan	24.59	24.59	24.46	24.59	24.40	24.58
Paris	31.90	31.90	31.75	31.89	31.88	31.88
Avg.	34.93	34.93	34.79	34.92	34.88	34.91
Diff.	-	0.00	0.14	0.02	0.06	0.02

TABLE IV. COMPARISON OF THE AVERAGE SPEED WITH THE CONVENTIONAL METHODS

	Speed Up					
	FS	**PDS**	**NPDS**	**APDS**	**TS-EPDS**	**Proposed**
Akiyo	1	10.78	14.57	34.95	183.14	400.98
Bridge_close	1	3.52	6.25	7.03	68.67	188.50
Children	1	8.24	14.38	29.45	90.64	152.66
Hall	1	3.54	14.14	16.09	41.59	158.89
Mother	1	3.65	13.66	14.83	66.02	187.80
News	1	7.95	14.45	28.86	120.92	232.93
Silent	1	6.59	14.37	27.36	84.06	176.96
Singer	1	10.58	14.55	33.88	170.21	337.43
Stefan	1	2.92	13.29	12.74	25.88	54.69
Paris	1	8.09	14.50	31.44	124.23	208.82
Avg.	1	6.58	13.42	23.66	97.54	209.97

ACKNOWLEDGMENT

"This work was supported by the National Research Foundation of Korea (NRF) Grant funded by the Korean Government (MOE) (NRF-2011-0011312).

REFERENCES

[1] Dufaux. F. and Moscheni. F.,“Motion estimation techniques for digital TV:a review and a new contribution,” Proc. IEEE, vol. 83,June 1995, pp. 858-876.

[2] JTC1/SC29/WG11, ISO/IEC 14496-2: Information Technology -coding of audio visual objects - Part 2: visual, (MPEG-4 Visual), 2000.

[3] JVT G050r1: Draft ITU-T recommendation and final draft internationalstandard of joint video specification (ITU-T Rec. h.264/ISO/IEC 14496-10 AVC), May 2003.

[4] J. Y. Tham, S. Ranganath, M. Ranganath, and A. A. Kassim, “A novelunrestricted center-biased diamond search algorithm for block motionestimation,” IEEE Trans. Circuits Syst. Video Technol., vol. 8, Aug. 1998, pp.369-377.

[5] L. M. Po and W. C. Ma, “A novel four-step search algorithm for fast block motion estimation,” IEEE Trans. Circuits Syst. Video Technol.,vol. 6, June 1996, pp. 313-317.

[6] R. Li, B. Zeng, and M. L. Liou, “A new three-step search algorithm forblock motion estimation,” IEEE Trans. Circuits Syst. Video Technol.,vol. 4, Aug. 1994, pp. 438-443.

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

[8] C. H. Cheung and L. M. Po, "Adjustable partial distortion search algorithm for fast block motion estimation," IEEE Trans. Circuits Syst. Video Technol., vol. 13, no. 1, Jan. 2003, pp. 100-110.

[9] M. G. Sarwer and Q. M. Jonathan Wu, "Efficient Two Step Edge based Partial Distortion Search for Fast Block Motion Estimation," IEEE Trans. Consumer Electron., vol. 55, no. 4, Nov. 2009, pp. 2154-2162.