



# Fast Motion Estimation Based on a Modified Median Operation for Efficient Video Compression

Jongho Kim\*, Member, KIICE

Department of Multimedia Engineering, Suncheon National University, Suncheon 540-950, Korea

## Abstract

Motion estimation is a core part of most video compression systems since it directly affects the output video quality and the encoding time. The full search (FS) technique gives the highest visual quality but has the problem of a significant computational load. To solve this problem, we present in this paper a modified median (MMED) operation and advanced search strategies for fast motion estimation. The proposed MMED operation includes a temporally co-located motion vector (MV) to select an appropriate initial candidate. Moreover, we introduce a search procedure that reduces the number of thresholds and simplifies the early termination conditions for the determination of a final MV. The experimental results show that the proposed approach achieves substantial speedup compared with the conventional methods including the motion vector field adaptive search technique (MVFAST) and predictive MVFAST (PMVFAST). The proposed algorithm also improves the PSNR values by increasing the correlation between the MVs, compared with the FS method.

**Index Terms:** Diamond Search, Fast motion estimation, Median predictor, MVFAST, PMVFAST

## I. INTRODUCTION

The block matching algorithm (BMA) is an essential part of many video coding standards, such as MPEG-1, 2, 4, and H.26x, to exploit the temporal correlation and reduce the redundancy that exists between frames of video sequences [1, 2]. In BMA, the current frame is partitioned into nonoverlapped blocks, and the best match for these blocks is found inside the search range of a reference frame with a predefined distortion criterion. The best matching block is then used as a predictor for the block in the current frame, where the displacement between the two blocks defines a motion vector (MV) of the current block. Thus, the encoder sends the MV and a residual block, which is the difference between the current block and the predictor, instead of the current block itself. This can significantly reduce the

number of bits involved in video coding procedures. The most common distortion measure is the mean absolute error or mean absolute difference, or equivalently, the sum of the absolute difference (SAD), which requires no multiplication and exhibits a performance similar to that of the mean squared error [3, 4]. Let  $f_i$  and  $f_{i-1}$  be a current frame and a reference frame, respectively. Then, for an  $M \times N$  block located at  $(x, y)$  in the current frame, the SAD is given by

$$SAD(d_x, d_y) = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} \left| f_i(x+m, y+n) - f_{i-1}(x+d_x+m, y+d_y+n) \right|, \quad (1)$$

where  $d_x$  and  $d_y$  denote the displacements in the  $x$ - and  $y$ -direction, respectively.

Received 14 June 2013, Revised 15 August 2013, Accepted 30 August 2013

\*Corresponding Author Jongho Kim (E-mail: [jhkim@suncheon.ac.kr](mailto:jhkim@suncheon.ac.kr), Tel: +82-61-750-3835)

Department of Multimedia Engineering, Suncheon National University, 255 Jungangno, Suncheon 540-950, Korea.

Open Access <http://dx.doi.org/10.6109/jicce.2014.12.1.053>

print ISSN: 2234-8255 online ISSN: 2234-8883

© This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Copyright © The Korea Institute of Information and Communication Engineering

The full search (FS) technique selects the MV, which points to the best matching block that has a minimum SAD, by exhaustive search at all possible points within the search range. Since the calculation of SAD at all possible locations is computationally intensive, it is difficult to directly use FS for various applications, such as digital TV, Internet streaming, and mobile video, in spite of its simple implementation. For this reason, many fast search algorithms have been suggested in an effort to reduce the computational complexity by searching only a subset of the possible search points, such as two-dimensional (2D) log search [5], three-step search (TSS) [6], new three-step search (NTSS) [7], and diamond search (DS) strategies [8, 9]. However, these methods usually incur a significant loss in the PSNR, although they can make the search faster by calculating the SAD less frequently. Moreover, these algorithms are not exactly optimal in consideration of the bits required to encode each MV, since they find MVs just by minimizing the SAD irrespective of the distribution of MVs.

In particular, the motion vector field adaptive search technique (MVFAST) and the predictive MVFAST (PMVFAST) have been introduced in order to improve the DS technique without sacrificing PSNR [10-12]. The MVFAST considers the MVs of the spatially adjacent blocks and the PMVFAST includes the MVs of the spatially and temporally adjacent blocks as possible candidates. The priority of the initial search point in PMVFAST is given to results of the median operation for the candidates, whereas priority in DS is given to a (0, 0) motion vector,  $MV_0$ . In addition, the early termination by a thresholding technique in the PMVFAST further reduces the computational complexity of the search procedures. These techniques of PMVFAST allow the video encoders to generate fewer bits for MVs with a negligible PSNR loss as well as significantly reduce the computations for search procedures, since the relatively high correlation between the MVs can be used effectively.

In this paper, we propose a modified median (MMED) operation and effective early termination conditions for the fast motion estimation. Since the MV of a temporally co-located block is more correlated with the final MV in many cases, our approach uses the MVs of four adjacent blocks, which include the left, top, and top-right blocks of the current block and a temporally co-located block in a reference frame, whereas PMVFAST uses the MVs of three spatially adjacent blocks. To select an appropriate candidate in the initial search procedure, we introduce an MMED operator that can decrease the number of search points, and this reduces the overall computational load. Since the initial search point is chosen more appropriately, thresholds and early termination conditions can be simplified, and consequently, we achieve faster motion estimation without sacrificing PSNR compared with the case of PMVFAST. The rest of this paper is organized as follows: Section II

presents a review of MVFAST and PMVFAST. In Section III, the proposed fast motion estimation technique is discussed in detail. Experimental results and conclusions are given in Sections IV and V, respectively.

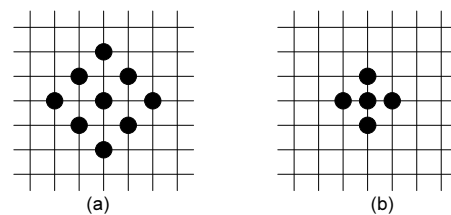
## II. REVIEW OF MVFAST AND PMVFAST

Most of the fast motion estimation algorithms first choose several candidates and then, calculate SAD at the candidate points. After the selection of the candidate with the minimum SAD, more search points around the selected candidate are tested. If the best candidate is not found in this procedure, the current candidate becomes the final MV. Therefore, the efficiency of the fast search algorithms depends on the determination of the candidates. In TSS, the number of search points does not change significantly according to the number of moving objects in the video sequences. On the other hand, in the DS that is biased to  $MV_0$ , better performance can be observed when the video sequences contain a few moving objects. MVFAST and PMVFAST also use DS patterns for fast motion estimation that is adaptive to the characteristics of video sequences. We will now briefly describe MVFAST and PMVFAST, which improve the search speed significantly with a negligible PSNR loss, since the proposed method is based on them.

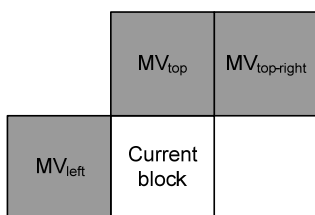
### A. Search Procedures of MVFAST

MVFAST introduced in [10, 12] improves the speed-up of the DS algorithm significantly by initially considering a small set of candidates, which includes  $MV_0$  and the MVs of three spatially adjacent blocks (the left, top, and top-right blocks). Unlike the DS that uses only the large diamond search pattern (LDSP) shown in Fig. 1(a), MVFAST considers the small diamond search pattern (SDSP) shown in Fig. 1(b).

Both the diamond patterns (LDSP and SDSP) keep on moving until the minimum SAD is found at the center. If the center point is selected using SDSP, the search procedure is terminated at this point. If the center point is selected using LDSP, the search procedure is carried out again using SDSP around the center of LDSP.



**Fig. 1.** Diamond search patterns: (a) large diamond search pattern (LDSP) and (b) small diamond search pattern (SDSP).



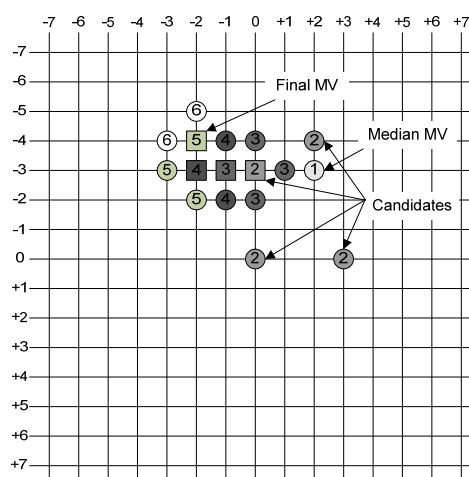
**Fig. 2.** Motion vectors of spatially adjacent blocks. MV: motion vector.

In MVFAST, the selection between SDSP and LDSP depends on the motion activity that the current block will most likely have. This motion activity is characterized by the magnitude of the largest MV among the three adjacent MVs, i.e.,  $MV_{left}$ ,  $MV_{top}$ , and  $MV_{top-right}$ , as shown in Fig. 2.

The motion activity is described as *low* when it is below a threshold  $L_1$ , *medium* when it is between  $L_1$  and a second threshold  $L_2$ , and *high* when it is larger than  $L_2$ . If the motion activity is considered *low*, SDSP is applied with  $MV_0$  as the center. If the motion activity is *medium*,  $MV_0$  is selected again as the center and LDSP is applied at this time instead of SDSP. This is the only case in which LDSP is selected, whereas the DS algorithm always chooses LDSP. After testing with LDSP, we apply SDSP again at the point with the smallest SAD as the center. Finally, if the motion activity is characterized as *high*, the center is determined with the position of the MV that yields the minimum SAD among  $MV_0$  and the spatially adjacent MVs, and then, SDSP is applied at this point. Moreover, in order to increase the search speed, MVFAST can introduce an early termination step, which is performed initially in the algorithm, with a fixed threshold (e.g.,  $T = 512$ ) at the  $MV_0$  position. When the SAD at the  $MV_0$  position is lower than  $T$ , MVFAST is stopped without examining any other locations.

### B. Search Procedures of PMVFAST

Unlike MVFAST whose search priority is given to  $MV_0$ , the priority of PMVFAST is given to the median of three spatially adjacent MVs [11, 12]. If SAD at the point obtained by the median operation is sufficiently small (e.g.,  $T = 256$ ), the search is terminated early after examining just one point. Otherwise, another termination condition is considered as follows: If the MV selected by the median operation is equal to the MV of a temporally co-located block ( $MV_{t-1}$ ) and SAD at this point is smaller than  $SAD_{t-1}$ , which is SAD between the block in  $f_{t-1}$  and the block pointed by  $MV_{t-1}$  in  $f_{t-2}$ , the search is stopped. When the algorithm is not terminated after checking the median operation, the final MV is searched by examining the five candidates,  $MV_0$ ,  $MV_{left}$ ,  $MV_{top}$ ,  $MV_{top-right}$ , and  $MV_{t-1}$ . After the selection of the best candidate from among them,



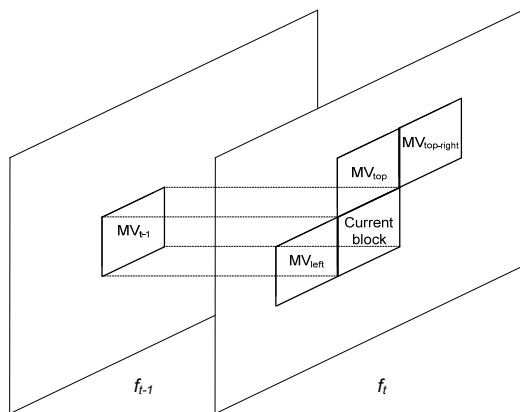
**Fig. 3.** An example of the search procedure for the best matching block in predictive motion vector field adaptive search technique. MV: motion vector.

a threshold  $T_1$  is applied; that is, the search is stopped if SAD calculated by the best candidate is smaller than  $T_1$ , which is selected adaptively as the smallest SAD among the SADs of the left, top, and top-right blocks. Otherwise, the best candidate is used as the center of the DS pattern. Furthermore, if the best candidate and  $MV_{t-1}$  are identical and SAD at the best candidate is smaller than  $SAD_{t-1}$ , the search is terminated.

In PMVFAST, LDSP is selected less frequently to reduce the total number of search points. To determine which DS pattern should be used, another threshold  $T_2$ , defined as  $T_1 + 256$  is applied. LDSP is chosen when  $T_2$  is larger than 1536 and the result of the median operation is  $MV_0$ , whereas SDSP is used in all other cases. When the smallest SAD is obtained at the center of the pattern, the algorithm is terminated. Additionally, PMVFAST has another early termination condition as follows: If all of the MVs of three spatially adjacent blocks are identical and  $MV_{t-1}$  is equal to the result of the median operation, the candidates (three spatially adjacent MVs and  $MV_0$ ) are highly correlated with each other. In this case, the selected search pattern is applied just once. Fig. 3 shows an example of searching the best matching block where the square shapes represent the points that have the smallest SAD in each step.

### III. PROPOSED MOTION ESTIMATION

We introduce a MMED operation to predict the appropriate starting search point, which significantly affects the performance of a fast motion estimation technique. Furthermore, the proposed algorithm reduces the number of thresholds and early termination conditions, and it consequently prevents the degradation in PSNR.



**Fig. 4.** Temporally co-located motion vector, where  $f_t$  and  $f_{t-1}$  represent the current frame and the reference frame, respectively. MV: motion vector.

### A. Modified Median Operation

PMVFAST feeds three spatially adjacent MVs to the median operator in order to determine the initial search point. However, our investigation presents that a temporally co-located MV,  $MV_{t-1}$  shown in Fig. 4, can be a good candidate for the starting search point, since it is highly correlated with the final MV.

Table 1 describes the average distances in pixel values between each candidate MV and the final MV as obtained by using PMVFAST. We can infer from the table that  $MV_{t-1}$  is more highly correlated to the final MV than the others in most of the test sequences. Therefore, we include  $MV_{t-1}$  in the median operation to determine the starting search point. Let  $(p_x, p_y)$  be a starting search point; then, the MMED operation is given by

$$\begin{cases} p_x = \frac{1}{2} \left\{ \sum_i p_x^i - \max_i(p_x^i) - \min_i(p_x^i) \right\}, \\ p_y = \frac{1}{2} \left\{ \sum_i p_y^i - \max_i(p_y^i) - \min_i(p_y^i) \right\}, \end{cases} \quad (2)$$

where  $i \in \{\text{left, top, top-right, } t-1\}$ , i.e.,  $p_x^i$  denotes the  $x$ -component of the starting search point selected among the MVs of the left, top, top-right, and temporally co-located blocks.

**Table 1.** Average distances in pixels between each candidate and the final MV in the case of PMVFAST

Sequence	$MV_{\text{left}}$	$MV_{\text{top}}$	$MV_{\text{top-right}}$	$MV_{t-1}$
Bus	1.44806	1.64125	1.88135	<b>0.81642</b>
Coastguard	0.73289	0.63438	0.73778	<b>0.57917</b>
Foreman	1.39627	<b>1.29688</b>	1.63483	1.53917
Stefan	2.25219	2.50256	2.79508	<b>2.08515</b>
Table Tennis	0.72754	0.71724	0.88137	<b>0.57195</b>

MV: motion vector, PMVFAST: predictive motion vector field adaptive search technique.

The proposed MMED operation can deal with even-numbered elements, whereas the conventional median operation used in PMVFAST can process only odd-numbered elements. A median operation eliminates the uncorrelated values more effectively than an average operation. On the other hand, the proposed MMED operation reduces the prediction error by averaging the spatially and temporally adjacent MVs and excluding the maximum and minimum values among them, which are considered to be less correlated to the final MV.

The starting search point for the current block at the typical position is determined as follows: When the current block is located at the origin (i.e., top-leftmost position) of the frame,  $MV_{t-1}$  is chosen directly as the predicted MV, since it is the only available candidate MV. When the current block lies in the first row except at the origin of the frame, two candidates,  $MV_{\text{left}}$  and  $MV_{t-1}$ , are available. Since the proposed MMED operator cannot deal with two elements,  $MV_0$  is added in this case. When the current block lies in the first column, except at the origin of the frame, or in the rightmost column, three candidates are available ( $MV_{\text{top}}$ ,  $MV_{\text{top-right}}$ , and  $MV_{t-1}$  for the first column, or  $MV_{\text{left}}$ ,  $MV_{\text{top}}$ , and  $MV_{t-1}$  for the rightmost column) and the starting search point is selected by the median rule instead of (2).

After the starting search point is determined, the early termination condition is applied. That is, if SAD at the point from (2) is less than  $T$  (e.g., 256), the search procedure is terminated. Furthermore, when the point from (2) is equal to  $MV_{t-1}$  and SAD at this point is smaller than  $SAD_{t-1}$ , the search procedure is also stopped. Table 2, which describes the frequency of the final MV determined after only one point is examined, presents that the MV from the proposed MMED operation is selected as the final MV more frequently than in the case of PMVFAST. This implies that the proposed algorithm can select the best matching block faster with a lower PSNR loss than PMVFAST.

### B. Determination of the Final Motion Vector

PMVFAST examines five candidates including  $MV_0$  after the selection of the starting search point by the median operation. In fact, as shown in Table 3,  $MV_0$  is rarely selected as the final MV when all the other candidates are

**Table 2.** The number of final MVs after the examination of only one point

Sequence	PMVFAST	Proposed algorithm
Bus	13542	15391
Coastguard	27941	32403
Foreman	24660	27205
Stefan	23572	24513
Table Tennis	46101	47066

MV: motion vector, PMVFAST: predictive motion vector field adaptive search technique.

**Table 3.** Frequency and ratio that  $MV_0$  is selected when all of the spatially and temporally adjacent MVs are not equal to  $MV_0$ 

Sequence	Frequency (times)	Ratio (%)
Bus	1282	0.023
Coastguard	3472	0.031
Foreman	2828	0.026
Stefan	4540	0.041
Table Tennis	1178	0.011

MV: motion vector.

not equal to  $MV_0$ . This implies that there is at least one  $MV_0$  among the spatially and temporally adjacent MVs when  $MV_0$  is selected as the final MV in most cases. Thus,  $MV_0$  does not need to be explicitly included in the search procedure, since it is possibly included in the spatially and temporally adjacent MVs.

Consequently, the proposed algorithm includes just four candidates of  $MV_{left}$ ,  $MV_{top}$ ,  $MV_{top-right}$ , and  $MV_{t-1}$  in the search procedure for the final MV. Since  $MV_0$  is not tested, the total number of search points is reduced and the correlation between MVs can increase. After an examination of these four candidates, the point that has the smallest SAD is selected as the center of SDSP. This further reduces the number of search points, since the proposed algorithm does not use LDSP. When the MVs are more correlated with each other in a video frame, the number of bits needed to encode each MV is reduced, since the video encoders usually encode the difference between the current MV and the adjacent MVs. We apply an early termination condition to the selected MV in order to further increase the correlation between the MVs as follows: A threshold  $T_1$ , which is limited between 512 and 1024, is set to the smallest SAD among the three spatially adjacent MVs ( $MV_{left}$ ,  $MV_{top}$ , and  $MV_{top-right}$ ). If SAD at the selected MV is smaller than  $T_1$ , the search procedure is stopped. When the selected MV is equal to  $MV_{t-1}$  and SAD at this point is smaller than  $SAD_{t-1}$ , the search is also terminated. For the other cases, the proposed search procedures apply SDSP to find the smallest SAD. In this case, the thresholds and early termination conditions are no longer used.

#### IV. EXPERIMENTAL RESULTS

The performance of the proposed method was evaluated on the basis of [13] by using the MPEG-4 verification model encoder [14]. Test sequences of QCIF ( $176 \times 144$ ), CIF ( $352 \times 288$ ), and SIF ( $352 \times 240$ ) were coded as IPPP; that is, all frames were inter-coded and only the first frame was intra-coded. TM5 was adopted as the rate control algorithm. Thresholds for motion activity in MVFAST,  $L_1$  and  $L_2$  were set to 1 and 2, respectively, and the threshold for early termination in PMVFAST and the proposed method,  $T$  was set to 256.

**Table 4.** Results of the proposed algorithm in terms of average PSNR and speed-up ratio

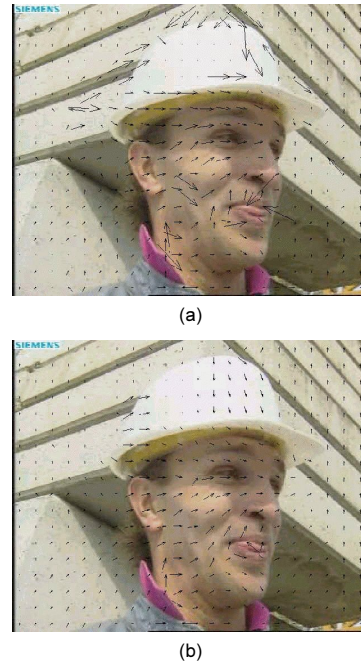
Measure	FS	MVFAST	PMVFAST	Proposed algorithm
PSNR (dB)	31.11	31.10	31.11	31.19
Speed-up	1.0	410.37	600.46	652.42

FS: full search, MVFAST: motion vector field adaptive search technique, PMVFAST: predictive MVFAST, PSNR: peak signal-to-noise ratio.

The maximum displacements of the search range were set to  $\pm 16$ ,  $\pm 32$ , and  $\pm 48$  pixels, respectively, in both the horizontal and the vertical direction.

Table 4 summarizes the performance of the proposed algorithm compared with MVFAST and PMVFAST in terms of both the PSNR and the speed-up ratio on average. The speed-up ratio is obtained by comparing the number of search points of each method with that of FS. Detailed results of the proposed approach and conventional methods for each experimental condition are described in Table 5.

It can be inferred from Tables 4 and 5 that the proposed algorithm significantly reduces the number of search points compared with those of conventional methods and exhibits similar or higher PSNRs than those of the FS. When the correlation between adjacent MVs increases, the video encoder can reduce the number of bits needed to encode each MV, and thus, the extra bits can be assigned for increasing PSNR. Fig. 5 shows the distribution of MV fields for the *Foreman* sequence from the FS and the proposed method. As shown in Fig. 5, the final MVs of the proposed algorithm are distributed more uniformly than those of the FS.

**Fig. 5.** Distribution of motion vector fields of the *Foreman* sequence for each search method: (a) full search and (b) proposed algorithm.

**Table 5.** Experimental results of the proposed algorithm compared with the conventional methods

Sequence (format, frame rate)	Bit-rate (kbps)	Search range	Measure	Motion estimation algorithm					Proposed algorithm
				FS	NTSS	DS	MVFAST	PMVFAST	
Hall Monitor (QCIF, 7.5 Hz)	10	16	PSNR (dB)	27.36	27.06	27.54	27.37	27.56	<b>27.64</b>
			Speed-up	1	46.16	64.62	211.93	317.22	<b>357.09</b>
		32	PSNR (dB)	27.28	27.02	27.55	27.35	27.62	<b>27.66</b>
			Speed-up	1	95.42	131.62	763.74	1125.91	<b>1294.16</b>
Mother & Daughter (QCIF, 10 Hz)	24	16	PSNR (dB)	<b>33.75</b>	33.64	33.64	33.54	33.64	33.72
			Speed-up	1	51.76	70.93	208.21	298.11	<b>310.14</b>
		32	PSNR (dB)	33.65	33.48	33.57	33.60	33.70	<b>33.70</b>
			Speed-up	1	168.71	241.92	757.16	1086.82	<b>1117.33</b>
Coastguard (QCIF, 10 Hz)	48	16	PSNR (dB)	28.52	28.26	28.34	28.51	28.51	<b>28.53</b>
			Speed-up	1	41.11	58.15	101.82	158.41	<b>182.54</b>
		32	PSNR (dB)	28.53	28.32	28.40	28.53	28.52	<b>28.55</b>
			Speed-up	1	139.65	157.81	367.26	564.70	<b>651.32</b>
Coastguard (CIF, 10 Hz)	112	16	PSNR (dB)	26.94	26.49	26.37	26.98	26.97	<b>26.99</b>
			Speed-up	1	36.11	47.82	98.20	111.25	<b>121.82</b>
		32	PSNR (dB)	26.96	26.53	26.40	27.00	27.03	<b>27.03</b>
			Speed-up	1	140.40	184.90	373.58	426.15	<b>464.30</b>
Foreman (CIF, 15 Hz)	512	16	PSNR (dB)	34.99	34.45	34.50	34.95	35.00	<b>35.00</b>
			Speed-up	1	40.25	47.29	98.51	139.87	<b>148.85</b>
		32	PSNR (dB)	35.17	35.31	34.38	35.11	35.17	<b>35.18</b>
			Speed-up	1	153.43	173.49	368.02	527.49	<b>563.45</b>
		48	PSNR (dB)	35.18	34.28	34.35	35.12	35.16	<b>35.18</b>
			Speed-up	1	306.53	327.92	789.30	1122.85	<b>1202.96</b>
Foreman (CIF, 30 Hz)	1024	16	PSNR (dB)	35.87	35.25	35.30	35.84	35.88	<b>35.88</b>
			Speed-up	1	44.56	55.70	127.71	205.57	<b>222.49</b>
		32	PSNR (dB)	38.09	37.35	37.42	38.03	38.08	<b>38.09</b>
			Speed-up	1	121.38	149.39	369.15	548.00	<b>585.42</b>
		48	PSNR (dB)	35.88	35.55	35.53	35.85	35.89	<b>35.89</b>
			Speed-up	1	372.26	390.81	1028.73	1646.50	<b>1779.69</b>
Table Tennis (SIF, 30 Hz)	1024	16	PSNR (dB)	<b>31.77</b>	31.60	31.70	31.73	31.75	31.75
			Speed-up	1	36.55	49.71	137.36	226.61	<b>251.86</b>
		32	PSNR (dB)	<b>31.76</b>	31.61	31.68	31.71	31.74	31.75
			Speed-up	1	139.08	187.86	516.55	848.29	<b>934.35</b>
		48	PSNR (dB)	<b>31.77</b>	31.61	31.71	31.69	31.73	31.74
			Speed-up	1	354.38	402.33	1093.70	1792.42	<b>1977.80</b>

FS: full search, NTSS: new three-step search, DS: diamond search, MVFAST: motion vector field adaptive search technique, PMVFAST: predictive MVFAST, PSNR: peak signal-to-noise ratio.

The proposed scheme is superior to PMVFAST, which is a widely used technique for fast motion estimation, in terms of both the speed-up ratio and the PSNR, as shown in Table 5. The performance of PMVFAST can be limited, since many thresholds without considering the characteristics of MVs possibly search for inappropriate points. The proposed algorithm searches faster than conventional algorithms and its average PSNRs are higher than those of the FS.

## V. CONCLUSIONS

This paper presents a fast motion estimation method based on a MMED operation and advanced strategies for the selection of the search points. The proposed MMED operation includes the temporally co-located MV for an initial candidate, which is more correlated with the final MV, as observed in our experiments. Moreover, we reduce the

number of thresholds and simplify the early termination conditions. With these improvements, the proposed approach achieves a faster search than MVFAST and PMVFAST. The proposed algorithm also improves PSNRs compared with those of the FS, since the MVs obtained from our approach are more correlated with each other and the bit-savings in coding them can be used for improvements in PSNR. Based on the better performance in terms of fast search and PSNR, the proposed algorithm can be employed in many applications, such as digital TV, Internet streaming, and mobile video by simply modifying the existing algorithms.

## REFERENCES

- [1] M. Ghanbari, "Principles of video compression," in *Standard Codecs: Image Compression to Advanced Video Coding*. London, UK: Institution of Electrical Engineers, pp. 25-61, 2003.

- [ 2 ] I. E. Richardson, *H.264 and MPEG-4 Video Compression*. Chichester, UK: John Wiley & Sons, 2003.
- [ 3 ] J. S. Kim and J. G. Kim, "A performance comparison of block-based matching cost evaluation models for FRUC techniques," *International Journal of Information and Communication Engineering*, vol. 9, no. 6, pp. 671-675, 2011.
- [ 4 ] G. Sorwar, M. Murshed, and L. Dooley, "Fast block-based true motion estimation using distance dependent thresholds," *Journal of Research and Practice in Information Technology*, vol. 36, no. 3, pp. 157-169, 2004.
- [ 5 ] J. R. Jain and A. K. Jain, "Displacement measurement and its application in interframe image coding," *IEEE Transactions on Communications*, vol. 29, no. 12, pp. 1799-1808, 1981.
- [ 6 ] T. Koga, K. Hirano, Y. Iijima, and T. Ishiguro, "Motion-compensated interframe coding for video conferencing," in *Proceedings of the National Telecommunications Conference*, New Orleans, LA, pp. G5.3.1-G5.3.5, 1981.
- [ 7 ] R. Li, B. Zeng, and M. L. Liou, "A new three-step search algorithm for block motion estimation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 4, no. 4, pp. 438-442, 1994.
- [ 8 ] S. Zhu and K. K. Ma, "A new diamond search algorithm for fast block matching motion estimation," *IEEE Transactions on Image Processing*, vol. 9, no. 2, pp. 287-290, 2000.
- [ 9 ] J. Y. Tham, S. Ranganath, M. Ranganath, and A. A. Kassim, "A novel unrestricted center-based diamond search algorithm for block motion estimation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8, no. 4, pp. 369-377, 1998.
- [10] P. I. Hosur and K. K. Ma, "Motion vector field adaptive fast motion estimation," in *Proceedings of the 2nd International Conference on Information, Communications and Signal Processing*, Singapore, pp. 7-10, 1999.
- [11] A. M. Tourapis, O. C. Au, and M. L. Liou, "Predictive motion vector field adaptive search technique (PMVFAST): enhancing block-based motion estimation," in *Proceedings of the Visual Communication and Image Processing*, San Jose: CA, pp. 883-892, 2001.
- [12] MPEG Video Group, "Optimized visual reference software," Text of 14496-7 PDTR, ISO/IEC N4057, 2001.
- [13] Implementation Study Group, "Experimental conditions for evaluating encoder motion estimation algorithms," ISO/IEC JTC1/SC29/WG11 N3141, 1999.
- [14] MPEG Video Group, "The MPEG-4 video verification model v.18.0," ISO/IEC JTC1/SC29/WG11 N3908, 2001.



#### Jongho Kim

received his B.S., M.S., and Ph.D. in Electronic Communications Engineering from Hanyang University, Seoul, Korea, in 1998, 2000, and 2008, respectively. From 2008 to 2009, he worked at Samsung Electronics Company and was involved in the development of video codecs and an IP-based video telephony system for mobile devices. He is currently an associate professor with the Department of Multimedia Engineering, Suncheon National University, Suncheon, Korea. His research interests include image and video compression, processing, and enhancement for multimedia applications, multimedia communication systems, and digital signal processing.