FAST MOTION ESTIMATION BY MOTION VECTOR MERGING PROCEDURE FOR H.264

Kai-Chung Hou, Mei-Juan Chen and Ching-Ting Hsu

Department of Electrical Engineering, National Dong-Hwa University, Taiwan

ABSTRACT

In this paper, a fast motion estimation algorithm for variable block-size by using a motion vector merging procedure is proposed for H.264. The motion vectors of adjacent small blocks are merged to predict the motion vectors of larger blocks for reducing the computation. Experimental results show that our proposed method has lower computational complexity than full search, fast full search and fast motion estimation of the H.264 reference software JM93 with slight quality decrease and little bit-rate increase.

1. INTRODUCTION

H.264[1] is a video compression standard being jointly developed by ITU-T Video Coding Experts Group and ISO/IEC Motion Picture Experts Group. The main goal of the standardization effort has been enhanced compression performance and provision of a network-friendly packet-based video representation. It can provide both objective and subjective image qualities superior to existing standards. The basic encoding algorithm of H.264 is similar to H.263 or MPEG standard except 4x4 integer transform instead of the traditional 8x8 DCT. Additional features including intra prediction mode, seven block-sizes for motion estimation (ME) and multiple reference frame selection are utilized in H.264 for higher coding efficiency.

In video coding standards, ME is a core functional block to remove the temporal redundancy in video sequences for achieving high compression. The tree-structured block-sizes of H.264 inter coded macroblock (MB) can be employed in the ME. Each MB can be coded by different block-modes including 16x16, 16x8, 8x16, and 8x8. If the 8x8 block-mode is chosen, each 8x8 block can be independently partitioned into 8x8, 8x4, 4x8, and 4x4 sub-blocks. So, altogether there can be seven different block-sizes: 16x16, 16x8, 8x16, 8x8, 8x4, 4x8, and 4x4. For these block-sizes, each MB contains 1, 2, 2, 4, 8, 8, and 16 motion vectors (MVs), respectively.

In the H.264 reference software (JM93) [2], a fast full search (FFS) algorithm and a fast ME (FME) are used for ME [3]. Recently, some fast variable block-size ME algorithms have been proposed. In [4], a fast search algorithm is applied to the seven block-sizes independently. In [5], a merging procedure is proposed for determining the MVs of the larger block-size from the MVs of the smaller block-size, which uses the threshold for the merging criteria related to the quantization parameter. In [6], the authors propose a fast method based on MVs' correlation to merge and split for ME. In [7], the low complexity merging procedure is proposed through the correlation of the neighboring blocks.

In this paper, we propose a ME algorithm by MVs merging procedure with a refinement method for variable block-size to reduce the computation.

The organization of this paper is shown as follows. In section 2, we describe the MV prediction and MV search refinement strategy. Experimental results are given in section 3. Finally, we conclude this paper in section 4.

2. FAST VARIABLE BLOCK-SIZE MOTION ESTIMATION BY MOTION VECTOR MERGING

2.1 Variable Block-Size MV Prediction

Both search center and search pattern are two important parts in a ME algorithm. In the previous ME algorithms, search center is generally predicted from the median, mean or the one with minimum SAD from the spatial or temporal neighbors' MVs [8][9]. However, this kind of methods assumes that the MV field is homogeneous. This assumption might fail if the video sequences have local motion and small moving objects.

Since the different block types are inside a MB, we can expect that the MVs of these blocks have high correlation. Hence, if we predict the search center of the current block from the MVs of the small blocks, it will be better than that from its neighboring MBs, especially when the motion field is not homogeneous.

To take advantage of the correlations with different block-sizes, the accuracy of the MV prediction is important. Because the motion vectors in small size blocks have high correlation and can stand for the real

0-7803-9332-5/05/$20.00 ©2005 IEEE

motion of the small object, the bottom-up merging will be utilized. Hence, we use the 4x4 block-mode as the initial block-size.

We observe the MVs' correlation of different block-sizes. The distribution of the differential MVs between the predicted MVs and the MVs by the full-search are investigated. The motion vectors of 4x4 blocks will be merged to 4x8, 8x4, 8x8, 8x16, 16x8 and 16x16 sequentially. They are shown in Fig. 1, and the predicted MVs is defined in (1). In addition, before using down-layer MVs as the source of predicted MVs, we perform a fast ME in section 2.2 to get more accurate MVs.

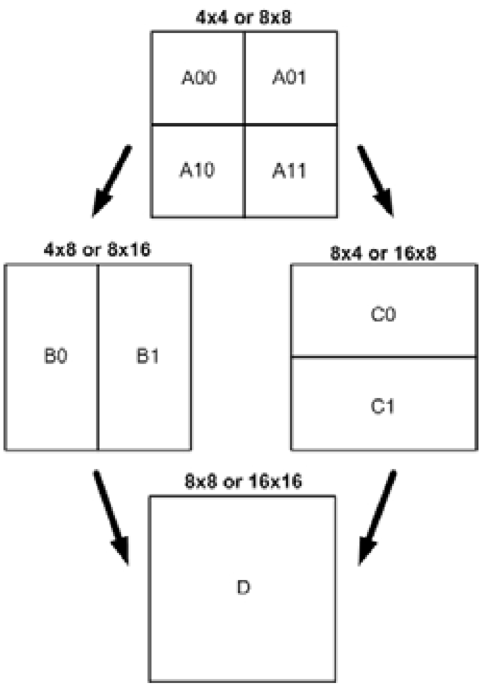

Fig. 1 Bottom up procedure by merging the MVs of the small blocks into the ones of the large blocks

$$
\begin{aligned}
PMV_{Bi} &= \frac{1}{2}(MV_{A0i} + MV_{A1i}), i = 0,1 \\
PMV_{Ci} &= \frac{1}{2}(MV_{Ai0} + MV_{Ai1}), i = 0,1 \\
PMV_{D} &= \frac{1}{4}\sum_{i=0}^{1}(MV_{Bi} + MV_{Ci})
\end{aligned}
\quad (1)
$$

PMV denotes the predicted MV by merging MVs in small blocks. And $MV_{A\sim D}$ denotes the MV of the corresponding block.

Let P(d) be the probability of MV difference(MV_{diff}) is less or equal to d as shown in (2).

$$P(d)_{HxV} = P(MV_{diff} <= d)_{HxV} \quad (2)$$

where the MV_{diff} denotes the difference of the MVs in small blocks and HxV denotes the block size.

2.2 Analysis of the Distribution of MVs

In [6-7], when the MVs in small blocks are the same, the MVs of the large blocks will be directly replaced by the MV of the small blocks. These methods will lose the accuracy because of no refinement for MV of the large blocks. From Table 1-Table 6, we compare the accuracy of the predicted MVs and the MVs by full search. In these tables, P(A|B) is the conditional probability of A, given that B has occurred, in which A refers to the predicted MVs in (1) with n-pel full search refinement, i.e., n=0,1,2; and B refers to the MV_{diff} in small blocks which the distance is less or equal to d. For example, P(1|0) shows the conditional probability of "the difference of the predicted MVs and MVs by full search in large block is equal or less to 1" given that the P(0) has occurred. And the P(0) is the probability that the MVs' difference in small blocks is equal to zero.

In these tables, we can observe that if we use the MV in small blocks as the MV in large blocks directly, such like P(0|0), it just takes 67.7% accuracy for 4x8 block of Foreman sequence. This is because optimal MVs in H.264 not only consider the sum of the difference (SAD) but also the bits of the coded MB for the optimal MVs. Therefore, if we use the MV in small blocks to predict the MV in larger size blocks and refine the MVs by +/- n-pel full search, we can get the MV for all the block-size modes with more accuracy.

Table 1 – The statistics of $P(d)_{4x8}$ and $P(A|B)_{4x8}$ (%)

Sequence	P(0)	P(0|0)	P(1|0)	P(2|0)
Carphone	56.7	76.5	85.4	88.1
Foreman	45.3	67.7	80.4	85.5
News	69.3	88.9	96.0	98.0

Table 2 -The statistics of $P(d)_{8x4}$ and $P(A|B)_{8x4}$ (%)

Sequence	P(0)	P(0|0)	P(1|0)	P(2|0)
Carphone	60.2	74.6	84.5	87.8
Foreman	48.2	65.2	77.9	83.7
News	69.6	85.4	94.8	96.2

Table 3 -The statistics of $P(d)_{8x8}$ and $P(A|B)_{8x8}$ (%)

Sequence	P(0)	P(0|0)	P(1|0)	P(2|0)
Carphone	46.9	72.9	88.9	92.3
Foreman	34.0	66.7	85.5	92.7
News	70.2	83.4	97.1	98.5

Table 4 - The statistics of $P(d)_{8x16}$and $P(A|B)_{8x16}$ (%)

Sequence	P(0)	P(0|0)	P(1|0)	P(2|0)
Carphone	28.7	81.8	89.3	90.7
Foreman	19.9	85.8	91.5	93.6
News	48.6	97.9	98.8	99.0

Table 5 -The statistics of $P(d)_{16x8}$ and $P(A|B)_{16x8}$ (%)

Sequence	P(0)	P(0|0)	P(1|0)	P(2|0)
Carphone	30.7	81.7	88.8	90.9
Foreman	21.2	82.4	89.3	92.2
News	46.6	96.7	98.4	98.7

Table 6 -The statistics of $P(d)_{16x16}$ and $P(A|B)_{16x16}$ (%)

Sequence	P(0)	P(0|0)	P(1|0)	P(2|0)
Carphone	37.3	63.0	90.8	95.0
Foreman	33.1	51.6	82.1	91.8
News	63.3	82.5	98.8	99.6

In addition, we analyze the distribution of the MVs in small blocks which are not identical. We analyze the conditional probability P(A|B). Let A be the predicted MVs with n-pel refining, i.e., n = 1, 2, 3, given that B is the difference of MVs in small blocks, the difference is 1 - 8. In Table 7, we show the conditional probability of prediction MVs in (1) with +/- 2-pel full search refinement, given that the distance of the MVs in the small blocks is less or equal to 5. We can observe that the accuracy of the prediction MVs is more than 80%.

Table 7– Statistics of average and refined MVs (%)

Sequence	4x8		8x4		8x8	
	P(2|5)	P(5)	P(2|5)	P(5)	P(2|5)	P(5)
Carphone	86.3	92.1	86.2	92.7	90.2	80.7
Foreman	81.3	90.8	81.2	91.3	88.4	82.7
News	97.1	97.4	95.8	97.1	96.5	91.5
Sequence	**8x6**		**16x8**		**16x16**	
	P(2|5)	P(5)	P(2|5)	P(5)	P(2|5)	P(5)
Carphone	90.8	80.9	90.8	80.9	88.8	87.9
Foreman	88.0	82.8	88.0	82.8	86.7	92.1
News	97.8	91.7	97.8	91.7	97.6	91.5

Thus, if we predict the MVs of large blocks in (1) with +/- n-pel refinement, we can get MVs for large blocks in high accuracy.

2.3 Proposed Fast ME Algorithm by Merge Procedure

As mentioned before, if we predict the MVs for large blocks in (1) with +/- 2-pel full search to refine the MVs, we can get more than 80% accuracy of the composed MVs. Therefore, we propose a fast ME algorithm by merging the MVs in small blocks with +/- 2-pel full search in larger blocks. The algorithm consists of two steps:

Step 1: initial MVs of the 4x4 block-size

We perform MDRPS (multi-directional rood pattern search)[10] for the 4x4 blocks ME since that MDRPS refers to multi-direction and achieves high PSNR with low computational complexity.

Step 2: Predict up-layer MVs from MVs of small blocks

if $|DMV_{small}| <= 5$

Here, DMV_{small} denote the difference of MVs for the small-block which are preformed the fast ME with MDRPS or refinement with +/-2-pel full search. We predict MVs in (1) as the MV for the larger-size blocks as illustrated in Fig. 1 and (1), then perform a +/-2-pel full search with this MV.

else

Perform the fast ME with MDRPS to get MV in larger block-size.

The flowchart is illustrated in Fig. 2. First, we utilize the fast ME algorithm, MDRPS, for 4x4 blocks. After finishing ME for block 4x4, we merge the MVs for large blocks. If the difference of the MVs in small blocks is less or equal to 5, we merge them in large blocks and refine the predicted MVs with a +/- 2-pel full search, otherwise, we perform MDRPS in large blocks. The algorithm is applied to multiple reference frames.

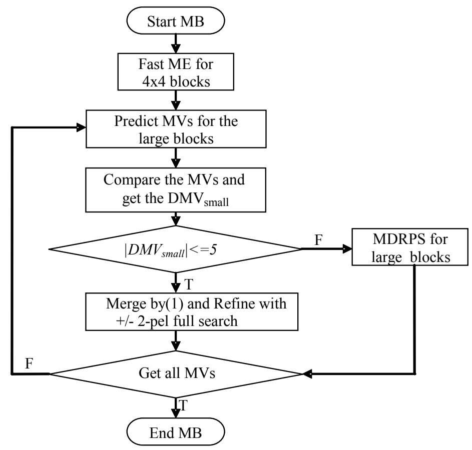

Fig. 2 The flowchart of the proposed algorithm

3. SIMULATION RESULTS

We implement our proposed algorithm in JM93 and compare the coding time, PSNR and bitrate of our proposed algorithm with the ones of full search (FS), fast full search(FFS), which re-uses SADs of 4x4 blocks to reduce computation, and fast motion estimation (FME), which uses hybrid unsymmetrical-cross multi-hexagon-grid search and early termination to seed up for ME in JM93. We test for seven sequences: Carphone, Foreman, Container and News QCIF sequences, and Mobile, Stefan and Weather CIF sequences. We calculate the coding gain with (3)-(6)

$$\Delta Ttime = \frac{Total\ time\ of\ proposal - Total\ time\ of\ FS\ in\ JM93}{Total\ time\ of\ FS\ in\ JM93} \cdot 100\% \quad (3)$$

$$\Delta MEtime = \frac{ME\ time\ of\ proposal - ME\ time\ of\ FS\ in\ JM93}{ME\ time\ of\ FS\ in\ JM93} \cdot 100\% \quad (4)$$

$$\Delta PSNR = PSNR\ of\ proposal - PSNR\ of\ FS\ in\ JM93 \quad (5)$$

$$\Delta Bitrate = \frac{Bitrate\ of\ proposal - Bitrate\ of\ FS\ in\ JM93}{Bitrate\ of\ FS\ in\ JM93} \cdot 100\% \quad (6)$$

The comparison of the total coding time is shown in Table 8. We can observe that our proposed algorithm is faster than other works. Furthermore, we compare the ME coding time in Table 9. Because our proposed algorithm has higher accuracy of the predicted MV, we have more coding efficiency than other works. Table 10 shows the comparison of the video quality. Our proposed algorithm just a little decrease to the full search. The total bit-rates are shown in Table 11. Our proposed algorithm just has a little increased bit-rate compared to other works. We utilize the MVs of the 4x4 blocks as the initial merge blocks, which stand for the real motion of the small object and the MVs of these blocks have high correlation. Therefore, we can accurately predict the MVs of the large blocks by using the MVs of small block with +/- 2-pel full-search refinement to reduce the huge computational complexity of the variable block-size motion estimation in H.264.

3. CONCLUSION

A fast variable block-size motion estimation algorithm is proposed in this paper. The proposed algorithm composes the MVs in variable block-size by using a bottom-up merging procedure with a +/- 2 full search refinement. The multi-directional rood pattern search, MDRPS, is used for motion estimation, and 4x4 blocks are as the initial stage. Experimental results show that our proposed algorithm has higher coding efficiency than the full-search ME, with slight quality decrease and low bit-rates increase.

Table 8 –The comparison of the total coding time (secs/150frames)

	Sequence	FS	Fast FS	Fast ME	Proposed	ΔTtime
QCIF	Carphone	987.8	885.1	422.7	409.9	-58.5%
	Foreman	1131.3	876.2	441.9	421.9	-62.7%
	Container	1092.4	832.1	474.6	443.4	-59.4%
	News	1110.0	897.3	428.8	418.4	-62.3%
CIF	Mobile	5572.3	3577.5	2184.2	2025.2	-63.7%
	Stefan	5058.3	3581.5	2006.7	1907.9	-62.3%
	Weather	3954.2	3577.5	1895.2	1700.2	-57.0%

Table 9 –The comparison of the ME coding time (secs/150frmaes)

	Sequence	FS	Fast FS	Fast ME	Proposed	ΔMEtime
QCIF	Carphone	595.7	437.7	268.6	214.9	-63.9%
	Foreman	694.0	434.2	281.1	243.7	-64.9%
	Container	592.4	430.2	255.8	213.3	-64.0%
	News	629.7	429.0	258.5	232.9	-63.0%
CIF	Mobile	2843.8	1449.4	921.3	873.2	-69.2%
	Stefan	3004.3	1502.5	984.7	927.0	-69.1%
	Weather	1835.0	1447.5	872.6	802.6	-56.3%

Table 10 –The comparison of the PSNR (dB)

	Sequence	FS	Fast FS	Fast ME	Proposed	ΔPSNR
QCIF	Carphone	37.39	37.39	37.36	37.34	-0.05
	Foreman	35.83	35.83	35.80	35.80	-0.03
	Container	36.08	36.08	36.08	36.07	-0.01
	News	36.82	36.82	36.78	36.77	-0.05
CIF	Mobile	34.33	34.33	34.32	34.30	-0.03
	Stefan	35.73	35.73	35.72	35.67	-0.06
	Weather	36.88	36.88	36.89	36.85	-0.03

Table 11–The comparison of the bit-rate (kbits/sec)

	Sequence	FS	Fast FS	Fast ME	Proposed	ΔBitrate
QCIF	Carphone	87.1	87.4	84.1	88.9	+2.1%
	Foreman	113.5	114.1	112.4	114.9	+1.2%
	Container	36.9	36.9	38.9	37.7	+2.2%
	News	76.0	75.8	71.1	77.82	+2.4%
CIF	Mobile	1584.4	1584.3	1581.9	1601.5	+1.1%
	Stefan	1014.8	1018.6	1018.4	1060.1	+4.5%
	Weather	167.4	167.0	167.3	168.5	+0.7%

4. REFERENCES

[1] Joint Video Team of ISO/IEC MPEG and ITU-T VCEG, "Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification," March 2003.

[2] Joint Video Team software JM93, August 2003. http://bs.hhi.de/~suehring/tml/download/

[3] Z. Chen, P. Zhou and Y. He, "Fast Motion Estimation for JVT," *JVT-G016, 7th Meeting, Pattaya II, Thailand*, March 2003.

[4] K.K. Ma and G. Qiu, "An Improved Adaptive Rood Pattern Search for Fast Block-Matching Motion Estimation in JVT/H.26L," *Proceeding of IEEE International Symposium on Circuits and Systems,* Vol. 2, pp. 708-711, Bangkok, May 2003.

[5] Y.K. Tu, J.F Yang, Y.N. Shen and M.T. Sun, "Fast Variable-Size Block Motion Estimation Using Merging Procedure with an Adaptive Threshold," *Proceeding of IEEE International Conference on Multimedia & Expo*, Vol. 2, pp. 789-792, 2003.

[6] Z. Zhou, M.T. Sun, and Y.F. Hsu, "Fast Variable Block-Size Motion Estimation Based on Merge and Split Procedures for H.264/MPEG-4 AVC," *Proceeding of IEEE International Symposium on Circuits and Systems*, Vol. 3, pp. 725-728, 2004.

[7] S.C. Tai, Y.R. Chen and S.J Li, "Low Complexity Variable-Size Block-Matching Motion Estimation for Adaptive Motion Compensation Block Size in H.264," *Proceeding of IEEE Asia-Pacific Conference on Circuits and Systems,* pp.613-616, 2004.

[8] Y. Nie and K.K. Ma, "Adaptive Rood Pattern Search for Fast Block-Matching Motion Estimation," *IEEE Transactions on Image Processing,* Vol.11, pp. 1442-1449, Dec. 2002.

[9] A. M. Tourapis, O. C. Au, and M. L. Liou, "Predictive Motion Vector Field Adaptive Search Technique (PMVFAST) Enhancing Block-based Motion Estimation," *Proceeding of SPIE Conference on Visual Communication and Image Processing*, pp. 883-892, Jan. 2001.

[10] G.L. Li, M.J. Chen, H.J. Li and C.T. Hsu, "Efficient Search and Mode Prediction Algorithms for Motion Estimation in H.264/AVC," *Proceeding of IEEE International Symposium on Circuits and Systems*, Kobe, Japan, May 2005.