Adaptive Variable Block-Size Early Motion Estimation Termination Algorithm for H.264/AVC Video Coding Standard

Mohammed Golam Sarwer and Q. M. Jonathan Wu, *Senior Member, IEEE*

***Abstract*—The variable block-size motion estimation (ME) process is the H.264/AVC encoder's most time-consuming function. This letter proposes to reduce the complexity of the ME process with an early termination algorithm that features an adaptive threshold based on the statistical characteristics of rate-distortion (RD) cost regarding current block and previously processed blocks and modes. In this method, most motion searches can be stopped early, with a large number of search points saved. A region-based search is also suggested to further reduce the computation required for full search ME. A search point reduction scheme for the fast motion estimation of H.264/AVC is also introduced, and the experimental results illustrate how the proposed method reduces ME times for full search and fast motion estimation by about 77 and 31%, respectively, despite the insignificant degradation of RD performance.**

***Index Terms*—H.264/AVC, motion estimation (ME), rate-distortion (RD) cost, video coding.**

I. INTRODUCTION

H.264/AVC is the latest video coding standard for the Joint Video Team (JVT), formed by ISO/IEC MPEG and ITU-T VCEG [1]. While H.264/AVC has clearly been a significant improvement in coding efficiency over previous standards, such as MPEG-1/2/4 and H.261/H.263, the computational complexity of H.264/AVCs motion estimation (ME) has also increased drastically. Various algorithms, such as three-step search [2], diamond search [4], and 2-D logarithm search [3], have been proposed to reduce the computational complexity of the ME module, and a number of first algorithms were developed for ME and mode decision [5]–[8], [20]–[23] in an attempt to accelerate H.264/AVC video coding. One way to reduce the complexity of the ME process is to terminate the ME calculation early in the process. At present, many attempts have been made to comprehensively explore the use of early termination algorithms in ME and mode decision for H.264/AVC video coding [9]–[15]. The improved fast-motion estimation method adopted by H.264/AVC reference software is presented in [14]. The variable block-size zero-motion detection and variable block-size best-motion detection algorithms compare rate distortion (RD) costs for two blocks [10], but the fixed thresholds developed in [10] are not sufficient to maintain high accuracy for all types of sequences and parameter settings.

This letter presents an efficient early termination method for the ME of H.264/AVC with early termination thresholds based on the RD cost of highly correlated blocks and modes. A region-based searching method is also provided for full search ME. An algorithm is also introduced to reduce fast-motion estimation search points adopted by H.264/AVC reference software.

The rest of this letter is organized as follows. Section II describes the algorithm for the proposed early termination method. Section III presents a statistical analysis of RD costs, and Section IV introduces the threshold selection scheme. Section V gives a region-based search order method for full search ME. Section VI provides an algorithm that reduces the search point of fast-motion estimation in H.264/AVC, while Section VII presents simulation results and Section VIII concludes this letter.

II. ALGORITHM OF PROPOSED EARLY TERMINATION

In H.264/AVC, the RD cost function J is defined as

$$J(mv, \lambda) = SAD(s, c(mv)) + \lambda R(mv - pmv) \tag{1}$$

where mv is the current motion vector (MV), pmv is the predicted MV, $SAD(s, c(mv))$ is the sum of absolute differences between current blocks and candidate block c for a given motion vector mv, λ is the Lagrangian multiplier, and R is the number of bits needed to code the MV. The objective of early termination is to decide whether a search point has met the RD cost criterion, so that the best search point for the current block can be found early. After searching each search point, the current RD cost J is compared against the threshold value of the specific block type, and the search is terminated if this RD cost J is lower than the threshold value. We define thresholds T_m $(m = 1, 2, \ldots, 7)$ for seven block sizes. If the cost of a search point satisfies (2), this search point is taken as the best one and the remaining searches can be skipped

$$J < T_m, \ \text{for } m = 1, 2, \ldots, 7. \tag{2}$$

III. STATISTICAL ANALYSIS OF RD COST FUNCTION

Our first series of experiments investigated whether or not previously calculated RD cost values could be used as a prediction metric for the early termination technique. Correlation

Manuscript received April 27, 2008; revised September 2, 2008. First version published April 7, 2009; current version published August 14, 2009. The work described in this paper has been supported in part by the Canada Research Chair Program, AUTO21 NCE, and the NSERC Discovery Grant. This paper was recommended by Associate Editor O. C. Au.

The authors are with the Department of Electrical and Computer Engineering, University of Windsor, Windsor, ON, Canada N9C2R7 (e-mail: sarwer@uwindsor.ca; jwu@uwindsor.ca).

Digital Object Identifier 10.1109/TCSVT.2009.2020322

TABLE I

(a) RD COST CORRELATION BETWEEN 16 × 16 MODE OF CURRENT MB AND 16X16 MODE OF CANDIDATE MB AND (b) RD COST CORRELATION BETWEEN CURRENT MODE AND PREVIOUSLY COMPUTED MODES

(a)

Sequence	Coll-ocated block in frame $t-1$	Coll-ocated block in frame $t-2$	Left	Upper	Upper right	Upper left
Stefan	0.95	0.90	0.74	0.75	0.58	0.61
Akiyo	0.98	0.97	0.49	0.71	0.44	0.38
Mobile	0.93	0.92	0.39	0.22	0.18	0.09
Foreman	0.90	0.83	0.51	0.36	0.21	0.23
Carphone	0.92	0.89	0.50	0.52	0.22	0.34
Salesman	0.98	0.96	0.65	0.39	0.31	0.25
Claire	0.98	0.97	0.59	0.82	0.42	0.51

(b)

Current mode	Previously computed mode						
	Coll-ocated MB in previous frame	Current MB in current frame					
	16 × 16	16 × 16	16 × 8	8 × 16	8 × 8	8 × 4	4 × 8
16 × 8	0.83	0.89	—	—	—	—	—
8 × 16	0.83	0.88	0.79	—	—	—	—
8 × 8	0.73	0.86	0.70	0.71	—	—	—
8 × 4	0.64	0.69	0.63	0.63	0.88	—	—
4 × 8	0.65	0.69	0.61	0.64	0.91	0.79	—
4 × 4	0.61	0.65	0.57	0.59	0.84	0.74	0.77

coefficients were calculated between RD cost values, using full search ME. The processing order for the different modes in the H.264/AVC ME process is 16 × 16, 16 × 8, 8 × 16, 8 × 8, 8 × 4, 4 × 8, and 4 × 4. This allows us to estimate the RD costs of lower size blocks from higher size blocks with the same macroblock (MB). In the case of the 16 × 16 mode, information about MV and RD costs for other modes with the same MB are not available, but the RD costs of neighboring MBs and collocated MBs in previous frames had already been computed.

Consider the current frame number to be t. Table I(a) shows the correlation coefficients of RD costs between the 16 × 16 mode of the current MB and the 16 × 16 mode of the candidate MBs. RD costs are clearly highly correlated with the collocated block in the immediately previous frame. The second highest correlation is with the collocated MB in frame $t-2$, which implies that the RD cost of the best search point for the 16 × 16 mode remains similar between consecutive frames and provides a good basis for predicting the RD cost of the current frame MB, which can then be used for early termination. The correlation of RD costs between other modes and previously computed modes are presented in Table I(b), which presents the average correlation coefficients of sequences used in Table I(a). Table I(b) reveals that the RD cost of the 16 × 8, 8 × 16, and 8 × 8 modes are highly correlated with the 16 × 16 mode of the same MB. Similarly, the RD costs of the 8 × 4, 4 × 8, and 4 × 4 modes are highly correlated with the 8 × 8 mode of the same MB, so the threshold value of lower size modes can be defined as a function of the best RD cost of either the 16 × 16 or the 8 × 8 block.

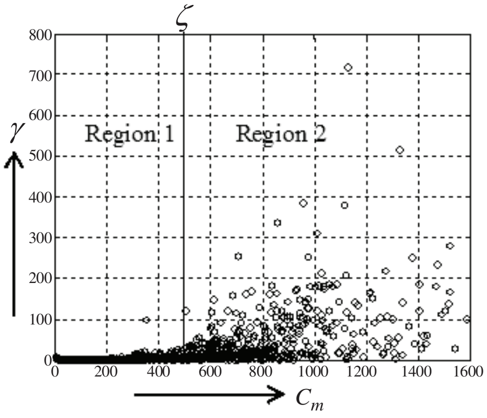


Fig. 1. Variation of γ with C_m at QP = 28 of *Foreman*.

IV. THRESHOLD SELECTION

The threshold for the 16 × 16 mode can be derived as a linear model of the RD costs of collocated MBs. Additionally, it is reasonable to say that a complex MB produces a large RD cost value compared to a simple MB. In the case of a complex MB, the difference in RD costs between the collocated MBs is also higher. These observations prompt the threshold for the 16 × 16 mode motion estimation to be defined as

$$T_1 = \frac{\alpha J_{t-1} + \beta J_{t-2}}{\alpha + \beta} + e\delta \tag{3}$$

where $\delta = |J_{t-1} - J_{t-2}|$ and α, β, e are the weighting factors. Here, J_{t-1} and J_{t-2} are the RD cost of the best MV for the 16 × 16 mode of collocated MB in frame $t-1$ and $t-2$, respectively. From some simulations, we have seen $\alpha = 3$, $\beta = 1$ and $e = 1/2$ produced better RD performance.

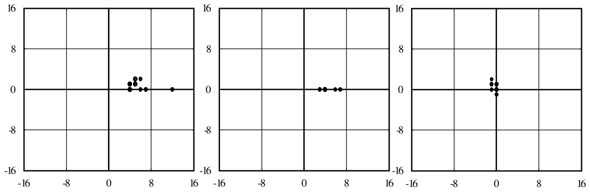
The thresholds for other modes are defined as

$$T_m = C_m + \gamma \quad \text{for } m = 2, 3, 4, 5, 6, \text{and } 7 \tag{4}$$

where, $C_m = (J_{16\times16}/S_m)$ for $m = 2, 3,$ and 4 (16 × 8, 8 × 16, and 8 × 8 modes) and $C_m = (J_{8\times8}/S_m)$ for $m = 5, 6,$ and 7 (8 × 4, 4 × 8, and 4 × 4 modes). The scale factor $S_m = 2$ for $m = 2, 3, 5, 6$ and $S_m = 4$ for $m = 4, 7$. γ is the penalty function and defined as

$$\gamma = |Actual \cos t - C_m|. \tag{5}$$

Fig. 1 shows the relationship between C_m and γ. The figure shows that the penalty function γ is very low, almost similar in region 1 ($C_m < \zeta$). The value of γ is increasing with C_m in region 2 ($C_m \geq \zeta$), which suggests that γ is approximated



(a) *Stefan* (6th frame: MB no: 40) (b) *Stefan* (6th frame: MB no: 2) (c) *Akiyo* (6th frame: MB no: 22)

Fig. 2. Forty-one Macroblock motion vectors.

as follows:

$$\gamma = A, \qquad \text{for } C_m < \zeta$$
$$\text{and } \gamma = BC_m + D, \quad \text{for } C_m \geq \zeta \tag{6}$$

where B is the slope of the best fitted straight-line and D is the constant. A is the mean value of γ within the range $C_m = 0$ to ζ. Simulations proved that the value of ζ increases from low motion sequences to high motion sequences, always within the range of 400 to 700. The *Foreman* sequence can represent scenes that possess relatively medium motion, which means that we can select $\zeta = 500$ based on *Foreman* and apply them to other sequences. In order to find out the value of A, B, and D, we have done extensive experiments with different video sequence types with different QP values. Better results were found at $A = 50$, $B = 1/8$, and $D = 45$.

V. REGION-BASED SEARCH ORDER FOR FULL SEARCH ME

The full search algorithm utilized in H.264/AVC applies a pixel-based spiral search method. But for high-motion sequences, the spiral search method is not more efficient [16]. This problem is addressed by dividing the search window into different regions and representing each region with a square of $N \times N$ size [16], with the size of the square related to the complexity reduction. Additionally, the early termination algorithm presented in the previous section is based on the inter-mode correlation, so in order to apply the region-based search, a modification is necessary. Fig. 2 displays the 41 motion vectors of a low-motion MB (*Akiyo*) and a high-motion sequence (*Stefan*). All the motion vectors are more directionally oriented for the high-motion sequences, so that dividing the search window into a square pattern is not as suitable as it is for low-motion sequences.

Based on this observation, we have divided the search window into 17 different regions as shown in Fig. 3. Region 0 represents the low-motion vectors. In this region, $|X| \leq 2$ and $|Y| \leq 2$. Regions 1–16 are based on the angle θ, which is defined as follows:

$$\theta = \tan^{-1}\frac{Y}{X}, \text{ where } |X| > 2 \text{ and } |Y| > 2. \tag{7}$$

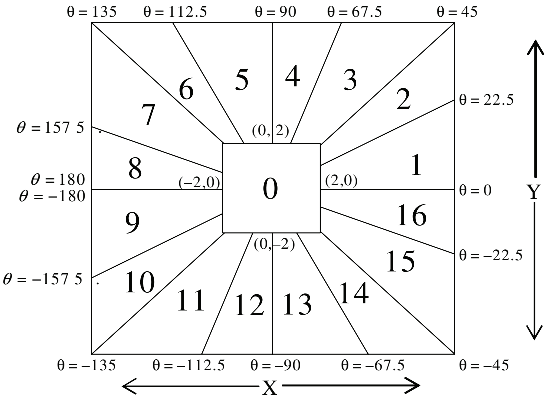


Fig. 3. Partitioned search range.

TABLE II
SELECTION OF MOST PROBABLE MV

Mode	Most probable motion vector
16 × 16 mode	Motion vector of the collocated macroblock in the previous frame
16 × 8, 8 × 16 and 8 × 8 mode	Motion vector of 16 × 16 mode
4 × 8, 8 × 4 and 4 × 4 mode	Motion vector of 8 × 8 mode

The goal of this algorithm is to reduce the number of search points in the early termination method, so the starting search region is a key factor of the resulting computation reduction. After the search range is partitioned, an adaptive search starts in the most probable region. Based on the statistical analysis presented in Section III, the most probable MV correspondence with different modes are tabulated in Table II. The most probable region is the region at which most probable MV falls.

The encoder starts to search from the most probable region and then moves to search region 0. The order of the remaining search regions is based on the angle between the most probable region and the other regions, which must be calculated by (8) beforehand

$$\Delta\theta = |\theta_m - \theta_R|,$$
$$\text{for } R = 1, 2, 3, \ldots, 16, R \neq \text{most probable region} \tag{8}$$

where θ_m is the direction of the most probable motion vector, and θ_R is the mean direction of region R, which are calculated as

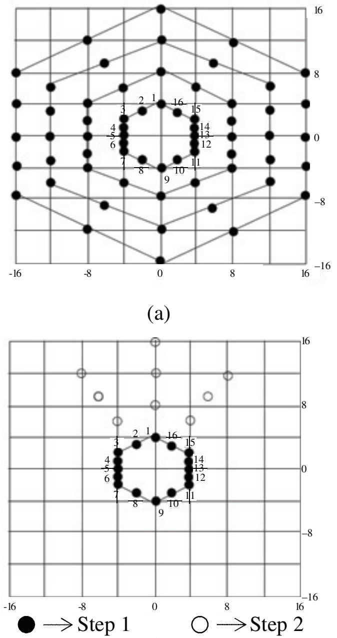
$$\theta_m = \tan^{-1}\left(\frac{MV_{my}}{MV_{mx}}\right) \tag{9}$$

$$\theta_R = \frac{\theta_{R(\max)} + \theta_{R(\min)}}{2} \tag{10}$$

where $\theta_{R(\max)}$ and $\theta_{R(\min)}$ are the maximum and minimum angle of region R. For example, in region 3 $\theta_{R(\max)} = 67.5^\circ$ and $\theta_{R(\min)} = 45^\circ$. MV_{mx} and MV_{my} are horizontal and vertical component of the most probable motion vector found in Table II. For example, the most probable region is 3 and $\theta_m = 50^\circ$, followed by a search order of 3, 0, 2, 4, 1, 5, 16, 6, 15, 7, 14, 8, 13, 9, 12, 10, and 11, which becomes slightly different when the most probable region is 0. In this case, the search starts from region 0 and the order of the remaining 16 search regions is calculated based on (8). The entire procedure of proposed region-based search with early termination is outlined in Fig. 4.

```
Divide the search region based on Fig. 3;
Assume the current macroblock is MB(x,y)
  For each mode m (m = 1, 2, …, 7) of the MB(x, y)
    Calculate the adaptive threshold, $T_m$;
    Calculate the search order of region;
      For each  block in the MB
        For each region
          For each search point
            If cost < $T_m$, (this search point is best point and
                go to the next  block);
            else    Do the normal motion search process;
            End if
          End for
        End for
      End for
  End for
  Set best mode to be the mode with the minimal cost
```

Fig. 4. Outline of proposed region based search with early termination.



(b)

Fig. 5. (a) Multihexagon search pattern and (b) proposed search pattern when point 1 is the best point in the inner hexagon.

VI. SEARCH POINT REDUCTION IN MULTIHEXAGON GRID SEARCH

A multihexagon grid search strategy is used for the fast integer pel motion estimation in JVT [6]. This algorithm has four different steps. In the third step, this method uses four different hexagons as shown in Fig. 5(a). The encoder searches all 64 points and the point with the minimum RD cost is chosen as the best motion vector. Some simulations showed that the rate (= 100 × *Number of best motion vector in a particular hexagon/Total number of best motion vector in all hexagon*) of the innermost hexagon was about 65%, decreasing from inner to outer hexagon.

As a result, the innermost hexagon was chosen as the most significant hexagon and became the first search point for all 16 points of this hexagon. It is reasonable to say that the RD costs of neighboring search points are similar. Assume that point 1 [indicated in Fig. 5(a)] is the best search point within the innermost hexagon. The best points of other hexagons should be one of the neighboring points of point 1. The proposed method considers this phenomenon and only searches the nine neighboring points of other hexagons. Fig. 5(b) shows the proposed search pattern. The proposed method first searches the 16 points in the inner hexagon to find the best point with a minimum RD cost, then the nine neighboring points to that best point are searched.

TABLE III

COMPARISON WITH FULL SEARCH MOTION ESTIMATION

Format	Sequence	ΔPSNR	ΔRate%	$\Delta T\%$	$\Delta SP\%$
QCIF (30 frames/s)	*Akiyo*	0.007	0.017	81.7	97.6
	Claire	0.005	0.138	80.3	97.2
	Carphone	0.011	0.353	74.3	92.0
	Coastguard	0.003	0.070	74.7	93.1
	Container	0.006	0.186	76.4	94.4
	Foreman	0.002	0.004	75.4	93.4
	Mobile	0.003	0.056	73.9	87.2
	Stefan	0.025	0.566	77.8	89.5
CIF (60 frames/s)	*News*	0.009	0.351	79.4	98.0
	Paris	0.013	0.529	76.1	94.2
	Bus	0.024	0.892	79.1	90.7
	Hall	0.003	0.707	77.7	96.7
Average		0.009	0.322	77.2	93.7

VII. SIMULATION RESULTS

To evaluate the performance of the proposed method, JM 9.6 [18] reference software is used in the simulations and different types of video sequences are used as test materials. The simulation conditions are: GOP structure IPPP, QP 16/20/24/28/31, number of reference frames one, RDO on, number of encoded frame 100, search range 16. Comparison results were produced and tabulated based on the average difference in ME time ($\Delta T\%$), average difference in search points ($\Delta SP\%$), the average PSNR differences ($\Delta PSNR$), and the average bit rate differences ($\Delta Rate\%$). PSNR and bit rate differences were calculated according to numerical averages between RD curves [19]. In order to evaluate the complexity reduction, $\Delta T(\%)$ and $\Delta SP\%$ are defined as follows:

$$\Delta T\% = \frac{T_{\text{original}} - T_{\text{proposed}}}{T_{\text{original}}} \times 100\% \qquad (11)$$

$$\Delta SP\% = \frac{SP_{\text{original}} - SP_{\text{proposed}}}{SP_{\text{original}}} \times 100\% \qquad (12)$$

where T_{original} denotes the ME time of the JM 9.6 encoder and T_{proposed} is the ME time of the encoder in the proposed early termination method. SP_{original} and SP_{proposed} are the average number of search points per MB of the original method and the proposed method, respectively.

A. Experiments With Full Search Motion Estimation

This experiment applies the proposed early termination (ET) method to the full search motion estimation of H.264/AVC. We chose twelve sequences with motion activities varying

TABLE IV
COMPARISON WITH FAST MOTION ESTIMATION

Format	Sequence	ΔPSNR	ΔRate%	ΔT%	ΔSP%
QCIF (30 frames/s)	*Stefan*	0.014	0.285	31.7	64.5
	Foreman	0.004	0.154	26.4	66.4
	Akiyo	0.002	0.052	27.8	69.4
	Mobile	0.001	0.029	32.4	62.0
	Carphone	0.006	0.184	29.9	66.2
	Claire	0.005	0.173	33.8	73.5
	Coastguard	0.009	0.205	30.4	67.9
	Container	0.005	0.130	35.1	74.9
CIF (30 frames/s)	*Foreman*	0.011	0.414	28.5	69.0
	Mobile	0.005	0.010	31.6	64.4
	Coastguard	0.002	0.077	35.4	71.1
	Container	0.009	0.027	29.0	68.0
	Flower	0.012	0.227	29.6	65.1
CIF (60 frames/s)	*News*	0.003	0.106	27.6	73.0
	Paris	0.008	0.203	31.1	71.7
	Bus	0.014	0.252	31.4	70.1
	Hall	0.008	0.357	37.0	72.9
Average		0.006	0.169	31.1	68.8

from small to large and the comparison results are tabulated in Table III, which shows how the proposed algorithm yields 77% ME time savings on average compared to a full search ME with a negligible PSNR reduction (0.009 dB) and bit rate increment (0.32%). Table III also lists the search point's comparison between the proposed method and the full search method. The reduction percentage of the search points (93%) is clearly not in accordance with the reduction percentage of the ME time (77%) because, in addition to block matching operations, there are other extra time-consuming aspects of fast algorithms (e.g., reading and writing memories and switching). For slow-motion sequences like *Akiyo*, *Claire*, and *News*, the proposed algorithm saves about 80% of ME time and 97% of search points. The computation reduction is high because most of the motion vectors for these types of sequences are around the search center, and the encoder gets the best point after searching only a few of the points.

B. Experiments With Fast Motion Estimation (FME)

This experiment applies our early termination method to the fast ME (FME) [6], [17] of H.264/AVC reference software, and the comparison results are tabulated in Table IV. FME is used as a benchmark, and Table IV shows how the computational complexity of the proposed scheme is much less than that of the FME. The proposed scheme saves about 31% of ME time and 69% of search points compared to FME. The rate distortion performance degradation is also very negligible. PSNR decrement is about 0.006 db, and the bit rate increment is about 0.16%. Table V compares the proposed method with the early termination methods described in [14]. The RD performance of the proposed method proves better for most sequence types, but for the complexity of a low-motion sequence like *Claire*, the other method is more efficient despite significant performance degradation. The proposed method is more efficient in terms of quality, bit rate, and complexity for medium- and high-motion sequences.

TABLE V
COMPARISON WITH OTHER METHOD AT 30 FRAMES/S

Sequence (QCIF)	ΔPSNR		ΔRate%		ΔSP%	
	Proposed	ET [14]	Proposed	ET [14]	Proposed	ET [14]
Stefan	0.014	0.057	0.285	1.16	64.5	32.1
Mobile	0.001	0.064	0.029	1.22	62.0	35.2
Claire	0.005	0.055	0.173	1.60	73.5	82.1
Coastguard	0.009	0.058	0.205	1.47	67.9	44.5
Average	0.007	0.058	0.173	1.36	67.0	48.5

VIII. CONCLUSION

This letter has developed a simple and fast early termination method for the ME of H.264/AVC. The adaptive thresholds are based on the correlation between the RD costs of current blocks and neighboring and collocated blocks. A method based on the region-based search order has been presented for a full search ME, and a fast algorithm for a multihexagon grid search also has been developed. The proposed method saves about 93 and 69% of search points, compared to H.264/AVCs typical full search and fast-motion estimations, respectively. The proposed algorithm can also be applied to any other conventional, fast-motion estimation method to further reduce its computation.

REFERENCES

[1] *Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification*, ITU-T Rec. H.264/ISO/IEC 14496-10 AVC, 2003.

[2] R. Li, B. Zeng, and M. L. Liuo, "A new three-step search algorithm for block motion estimation," *IEEE Trans. Circuits Syst. Video Tech.*, vol. 4, no. 4, pp. 438–442, Aug. 1994.

[3] J. R. Jain and A. K. Jain, "Displacement measurement and its application in inter frame image coding," *IEEE Trans. Commun.*, vol. COM-29, no. 12, pp. 1799–1808, Dec. 1981.

[4] S. Zhu and K.-K. Ma, "A new diamond search algorithm for fast block matching motion estimation," *IEEE Trans. Image Process.*, vol. 92, no. 2, pp. 287–293, Feb. 2000.

[5] Y.-K. Tu, J.-F. Yang, M.-T. Sun, and Y. T. Tsai, "Fast variable block motion estimation for efficient H.264/AVC encoding," *Signal Process.: Image Commun.*, vol. 20, no. 7, pp. 595–623, Aug. 2005.

[6] Z. Chen, P. Zhou, and Y. He, "Fast Integer Pel and Fractional Pel Motion Estimation for JVT," Joint Video Team (JVT) Docs, JVT-F017, Awaji, Japan, Dec. 2002.

[7] F. Pan, H. Yu, and Z. Lin, "Scalable fast rate-distortion optimization for H.264/AVC," *EURASIP J. Appl. Signal Process.*, vol. 2006, pp. 1–10, 2006.

[8] Y.-C. Lin, T. Fink, and E. Beller, "Fast mode decision for H.264 based on rate-distortion cost estimation," in *Proc. ICASSP-2007*, Honolulu, Hawaii, pp. 1137–1140.

[9] J. F. Yang, S. H. Chang, and C. Y. Chen, "Computation reduction for motion search in low rate video coders," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 10, pp. 948–951, Oct. 2002.

[10] L. Yang, K. Yu, J. Li, and S. Li "An effective variable block size early termination algorithm for H.264 video coding," *IEEE Trans. Circuits Syst. Video Tech.*, vol. 15, no. 6, pp. 784–788, Jun. 2005.

[11] S.-E. Kim and J.-K. Han, "Efficient motion estimation using a modified early termination algorithm in H.264," *IEICE Trans. Inf. & Syst.*, vol. E 88-D, no. 7, pp. 1707–1715, Jul. 2005.

[12] C.-Y. Su and S.-L. Chang, "Adaptive early termination for fast H.264 Video Coding," in *Proc. 9th IEEE Int. Symp. Multimedia 2007*, Taichung, Taiwan, pp. 72–76.

[13] G.-L. Li and M.-J. Chen, "Adaptive search range decision and early termination for multiple reference frame motion estimation for H.264," *IEICE Trans. Commun.*, vol. E-89B, no. 1, pp. 250–253, Jan. 2006.

[14] X. Xu and Y. He, "Improvement on fast motion estimation strategy for H.264/AVC," *IEEE Trans. Circuits Syst. Video Tech.*, vol. 18, no. 3, pp. 285–293, Mar. 2008.

[15] E. A. Al Qarallah and T.-S. Chang, "Fast variable block size motion estimation by adaptive early termination," *IEEE Trans. Circuits Syst. Video Tech.*, vol. 16, no. 8, pp. 1021–1026, Aug. 2006.

[16] S. Jin, S.-J. Park, and J. Jeong, "Adaptive fast full search algorithm using partitioned region and optimized search order," *IEEE Trans. Consum. Electron.*, vol. 53, no. 4, pp. 1703–1711, Nov. 2007.

[17] Z. Chen, P. Zhou, Y. He, and G. Wang, "Fast motion estimation for JVT," Joint Video Team, JVT-G016, Pattaya, Thailand, Mar. 2003.

[18] *JVT Reference Software version 9.6* [Online]. Available: http://iphome.hhi.de/suehring/tml/

[19] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," in *Proc. 13th VCEG-M33 Meeting*, Austin, TX, Apr. 2001.

[20] X. Sul, S. Singh, and Y. Bai, "Local reference with early termination in H.264 Motion estimation," in *Proc. ICME 2007*, Beijing, China, pp. 384–387.

[21] H. F. Ates and Y. Altunbasak, "Rate-distortion and complexity optimized motion estimation for H.264/AVC video coding," *IEEE Trans. Circuits Syst. Video Tech.*, vol. 18, no. 2, pp. 159–171, Feb. 2008.

[22] C.-M. Mak, C.-K. Fong, and W.-K. Cham, "Fast Motion Estimation for H.264/AVC in Walsh–Hadamard Domain," *IEEE Trans. Circuits Syst. Video Tech.*, vol. 18, no. 6, pp. 735–745, Jun. 2008.

[23] W. You, Y. Song, T. Ikenaga, and S. Goto, "A high quality fast motion estimation algorithm for H.264/AVC," in *Proc. 2008 Congr. Image Signal Process.*, Sanya, Hainan, pp. 375–379.