An Adaptive Search Range Selection Algorithm for HEVC

Yoshiki Ito[1],Wen Shi[1],Tian Song[2], and Takashi Shimamoto[2]
[1]College of Systems Innovation Engineering, Graduate School of Advanced Tcchnology and Science,
The University of Tokushima, Tokushima, 770-8506, Japan
Email: {yoshiki-ito,shiwen}@ee.tokushima-u.ac.jp
[2] Computer Systems Engineering, Institute of Technology and Science, Graduate School of Engineering,
The University of Tokushima, Tokushima, 770-8506, Japan
E-mail : {tiansong,simamoto}@ee.tokushima-u.ac.jp

Abstract: In this paper, we proposed an adaptive search range selection algorithm. In previous work, the method which determines the search range using motion vector variance was proposed. However, the selected search range is not efficient enough. The proposed algorithm can efficiently reduce the redundant search range using the distribution of the surrounding motion vectors. The simulation results show that the proposed algorithm achieved averagely 76.0% complexity reduction with 0.4% BD-rate increasing, compared with previous work.

Keywords—**Motion estimation, Search range, Complexity reduction, Coding efficiency, HEVC**

1. Introduction

Nowadays, the demand for high-definition video has catalyzed the increasing growth in of data volume for video. In this situation, high efficiency video coding (HEVC) which is the latest video compression standard was recommended in January 2013 by the joint collaborative team on video coding (JCT-VC). It achieved about twice compression ratio compared with H.264/AVC which is the previous video coding standard. HEVC contributes to high compression and high efficiency for coding video.

However the computation complexity in HEVC becomes an implementation bottle-neck. In HEVC, most of the complexity is occupied by motion estimation (ME). ME is an important coding tool adopted in almost all mainstream video compression standard such as MPEG-2, H.264/AVC and HEVC. ME significantly contributes to coding efficiency by removing the temporal data redundancy between frames. This is achieved by allowing blocks from currently coded frames to be matched with the ones from the reference frames. Only the blocks' differences along with a set of displacements need to be encoded in a process known as inter prediction [1].

ME process consists of two steps in HEVC. Firstly, it predicts the initial motion vector position, which is called the advanced motion vector prediction (AMVP). Then, ME searches the best matching block within the defined search area. There are various search patterns of ME such as three-step search [2], four-step search [3], diamond search [4], cross diamond search [5], hexagon-based search [6], etc. These methods can reduce the computation complexity of ME process in software-based encoders. However, it is difficult to realize these method in hardware implementation, therefore full search algorithm is used in hardware encoders. In order to find the best matching block, full search algorithm checks all candidates (search points) in the search area and consequently it involves huge computation complexity, since the rate-distortion (RD) cost of all the candidates within the search range is calculated. Thus, there is redundant computation complexity in ME process.

In previous work, an adaptive search range prediction algorithm was proposed to reduce computation complexity [7][8]. Their proposal determines search range (SR) based on motion vector (MV) variance of spatial and temporal neighboring blocks. This method achieved superior complexity reduction while obtaining an acceptable rate-distortion coding performance. However, this method is not perfect in computation complexity reduction. Since the variance σ^2 contains outlier, it can not imply the appropriate search range perfectly. For this reason, too large search range is selected. In addition, the distribution of the best point within the search range is not considered. In this work, we proposed an adaptive search range selection algorithm that modifies the redundant search range and predicts the distribution of the best point to reduce the redundant search area.

2. Adaptive search range selection based on motion vector variance

The probability of optimal MV within the chosen SR can be used to determine the tradeoff between the SR and the coding performance. The difference between the optimal MV and the MVP is called the motion vector prediction difference (MVPD), which is a random variable. If the distribution of the MVPD is concentrated in the zero with a small variance, a smaller SR can be adopted.

The method of previous work selects search range adaptively using variance of MV of spatial and temporal neighboring blocks (σ^2) to reduce computation complexity in ME. The variance σ^2 indicates the consistency of MV. Thus, when it is small, all MVs of neighboring blocks are similar. It is likely that optimal MV is close to MVP and smaller SR can be used. According to verification of previous work, when the variance of MV of neighboring blocks is small, it is highly probable that optimal MV is in the neighborhood of MVP. On the other hand, when the variance is larger, the distribution of optimal MV spreads over a wider range. In addition, the variance of MVPD grows with a larger value of the variance σ^2 [7]. A larger σ^2 means a larger variance of MVPD and demands a larger SR value.

Since the horizontal and vertical directions can be handled separately, the SR can be rectangular by using σ_x^2 and σ_y^2. In this method, the probability density function of MVPD was used to calculate the probability of optimal MV within the SR, which can be defined as

$$f_{MC,T}(t) = \frac{C_t}{(|t/\zeta_t|^{\frac{5}{3}} + 1)}, \quad t \in \{x, y\} \tag{1}$$

This is the modified zero-mean Cauchy density function with parameters ζ_x and ζ_y, and C_x and C_y are the normalization constants. The probability of optimal MV within (SR_x, SR_y) can be calculated via

$$F_{MC}(SR_x, SR_y) = F_{MC,X}(SR_x) \cdot F_{MC,Y}(SR_y) \tag{2}$$

$$F_{MC,T}(SR_t) = \int_{-SR_t-0.5}^{SR_t+0.5} f_{MC,T}(t)dt, \quad t \in \{x, y\} \tag{3}$$

Model parameters (ζ_x, ζ_y) can be derived from σ^2. It is possible to obtain proper values of SR_x and SR_y by defining the probability of the optimal MV to fall into the desired window.

In order to reduce computation complexity, the search range is selected by the look-up table. The look-up table is constructed using the unit variance. In case of σ^2, the probability of optimal MV within SR_t can be derived from $F_{MC}(t/\sigma^2)$. [7]

3. Proposed algorithm

As shown in Figure 1, our proposal uses the spatial and temporal neighboring blocks to calculate the variance of MVs. When the variance of spatial neighboring blocks is smaller than the variance of temporal neighboring blocks, we use the spatial variance because the spatial variance implies the necessary search range more accurately than the temporal variance. Thus we can select smaller search range with maintaining coding efficiency. In other cases, we use the mean value of the spatial variance and the temporal variance.

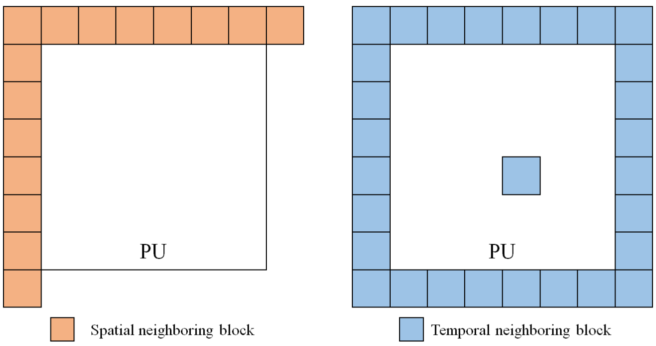

Figure 1. Neighboring blocks used in the variance calculation

3.1 Search range pre-definition

According to previous work, when variance σ^2 is large, MVPD is also large in most cases. However, sometimes too large search range is selected. Since the variance σ^2 contains outlier. It can not imply the appropriate search range perfectly. It leads to large computation complexity. Therefore, we define the search range using MVs of neighboring blocks. When the value of maximum difference (MaxMVD$_x$, MaxMVD$_y$) between MV of neighboring blocks and AMVP is smaller than the default search range, it has highly probability that the MVPDx and MVPDy are smaller than MaxMVDx and MaxMVDy respectively. Hence we define the search range to the difference (MaxMVD$_x$, MaxMVD$_y$). An example of the proposed search range is shown in Figure 2. In this case, there are MaxMVD$_x$ and MaxMVD$_y$ within the original search window. Therefore the proposed search range is defined as MaxMVD$_x$ and MaxMVD$_y$.

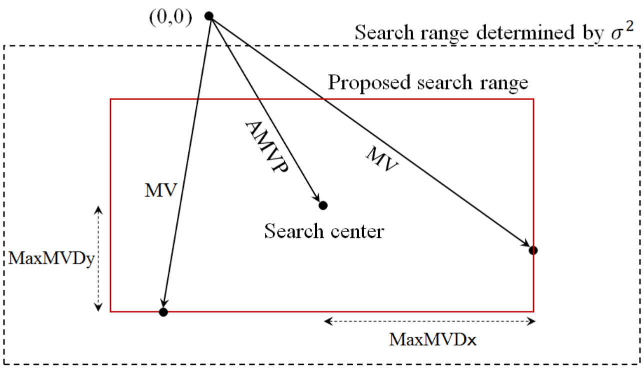

Figure 2. Modifying search range

3.2 Search range refinement

As shown in Figure 3, we define the refined search ranges by SR$_{x,+}$, SR$_{x,-}$, SR$_{y,+}$ and SR$_{y,-}$, based on the proposed search range. In previous work, left side SR$_{x,-}$ and right side SR$_{x,+}$ are determined at the same value using variance σ_x^2. Likewise, upper side SR$_{y,+}$ and lower side SR$_{y,-}$ are also determined at the same value. However, in many cases the optimal MV concentrated on one side. Our proposed algorithm predicts the distribution of the best point using the distribution of the MVs of the neighboring blocks to reduce the redundant search area, and refine the search range. Using following equations the proposed search range refinement algorithm will be introduced. The number of vectors which is located to right of the search center is denoted by N$_{right}$. Likewise, the number of vectors are denoted by N$_{left}$, N$_{above}$ and N$_{below}$ respectively. Besides, the search range which is derived by chapter 3.1 are denoted by SR$_{def,x}$ and SR$_{def,y}$.

when N$_{right}$ < N$_{left}$,

$$SR_{x,+} = \frac{N_{right}}{N_{letf}} \times SR_{def,x} \qquad SR_{x,-} = SR_{def,x} \tag{4}$$

when N$_{right}$ ≥ N$_{left}$,

$$SR_{x,-} = \frac{N_{left}}{N_{right}} \times SR_{def,x} \qquad SR_{x,+} = SR_{def,x} \tag{5}$$

when N$_{above}$ < N$_{below}$,

$$SR_{y,+} = \frac{N_{above}}{N_{below}} \times SR_{def,y} \qquad SR_{y,-} = SR_{def,y} \tag{6}$$

when N$_{above}$ ≥ N$_{below}$,

$$SR_{y,-} = \frac{N_{below}}{N_{above}} \times SR_{def,y} \qquad SR_{y,+} = SR_{def,y} \tag{7}$$

As an example in the case of Figure 3, equation (5), (7) are used to reduce the search area.

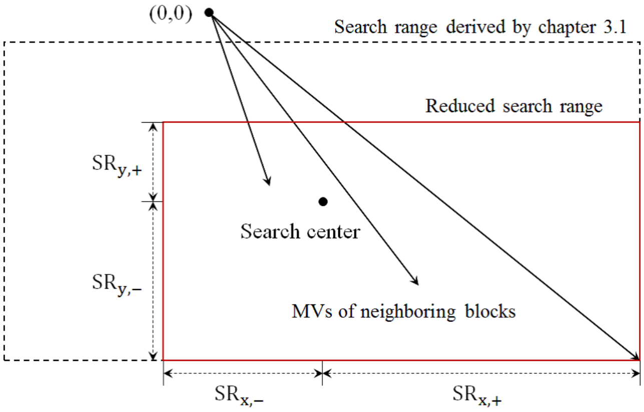

Figure 3. Distribution prediction

3.3 Flowchart of the adaptive search range selection

An adaptive search range selection algorithm which can improve the previous work is proposed. Our method improves unreasonable search range that is determined by the method of previous work. Then it predicts the distribution of the best point in the search area and reduces redundant search area.

A detailed flowchart of the proposal is described in Figure 4. Firstly, SR$_x$ and SR$_y$ are determined using σ_x^2 and σ_y^2 respectively. Secondly, when the value of maximum difference (MaxMVD$_x$, MaxMVD$_y$) between MV of neighboring blocks and AMVP is smaller than the search range, the search range is defined as the difference (MaxMVD$_x$, MaxMVD$_y$). Finally, our proposal checks the MV distribution of nieghboring blocks and reduces the redundant search area. At horizontal search range, when N$_{right}$ < N$_{left}$, equation (4) is used to reduce the right-side search area. In other case, equation (5) is used. At vertical search range, when N$_{above}$ < N$_{below}$, equation (6) is adopted to reduce the upper side search area. In other case, equation (7) is adopted.

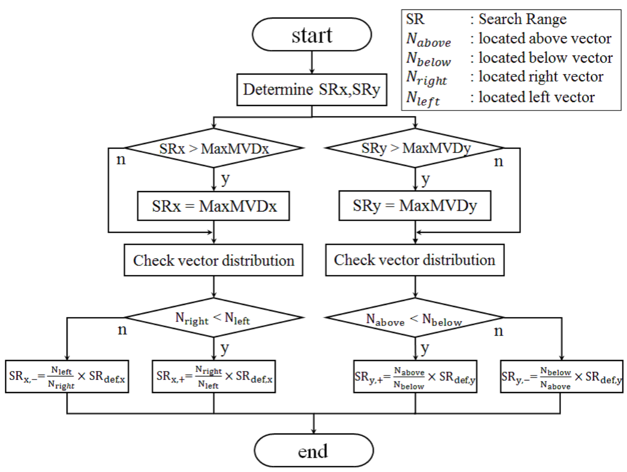

Figure 4. Adapted search range selection process

4. Simulation Results

The proposed adaptive search range selection algorithm is implemented in the HEVC reference software HM16.7. We configure the simulations with random access. Moreover, we set the probability of optimal MV within SR to 91% in the similar way of previous work [7]. The maximum search range value and the minimum search range value are defined by 64 and 2 respectively. Several test sequences (50 frames) with picture size of Full-HD are used. The computation complexity reduction is evaluated with different QP of 22, 27, 32, 37. Bitrate and PSNR of the proposed algorithm are compared with previous work by

$$\Delta Bitrate = \frac{Bitrate_{Pro} - Bitrate_{Pre}}{Bitrate_{Pre}} \times 100\,[\%] \qquad (8)$$

$$\Delta PSNR = PSNR_{Pro} - PSNR_{Pre}\,[dB] \qquad (9)$$

The computation complexity reduction is defined as

$$\frac{SearchPoints_{Pre} - SearchPoints_{Pro}}{SearchPoints_{Pre}} \times 100\,[\%] \qquad (10)$$

Table 1 shows the comparison of previous work [7] and the proposed algorithm. As shown in Table 1, the proposed algorithm achieved averagely 76.0% complexity reduction with 0.4% BD-rate increasing. Even in worst case (BasketballDrive), the proposed method achieved averagely 62.7% complexity reduction with 0.7% BD-rate increasing. The coding efficiency is evaluated by RD curve (Figure 5 to Figure 9). Experimental results show that the proposed algorithm can reduce computation complexity significantly with negligible coding RD performance degradation.

Table 1. Previous work vs. Proposed algorithm

Sequence	QP	Δ PSNR [dB]	Δ Bitrate [%]	BD-rate [%]	complexity reduction [%]
Kimono	22	0.0005	0.175	0.5	76.9
	27	0.0015	0.356		78.5
	32	-0.0067	0.383		78.6
	37	-0.0093	0.676		77.9
ParkScene	22	0.0014	0.168	0.3	85.8
	27	-0.0049	0.034		86.6
	32	-0.0069	0.291		85.9
	37	-0.0090	0.276		84.8
Cactus	22	0.0014	-0.078	0.2	68.4
	27	-0.0006	0.096		71.2
	32	-0.0059	0.082		71.8
	37	-0.0086	0.028		72.7
BasketballDrive	22	0.0002	0.118	0.7	54.4
	27	0.0021	0.347		62.4
	32	-0.0053	0.874		65.7
	37	-0.0140	0.629		68.4
BQTerrace	22	-0.0034	-0.001	0.3	82.6
	27	-0.0069	-0.050		84.6
	32	-0.0105	-0.080		82.9
	37	-0.0008	-0.019		79.0
Average		-0.0043	0.215	0.4	76.0

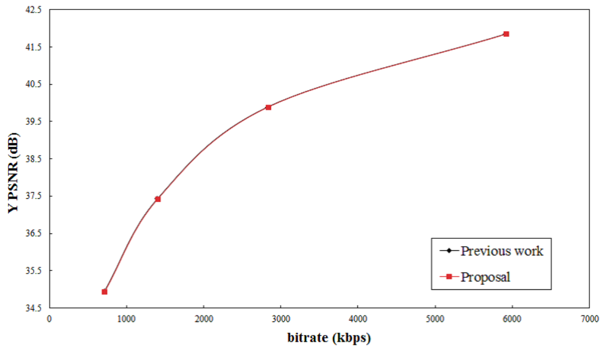

Figure 5. RD curve: Kimono

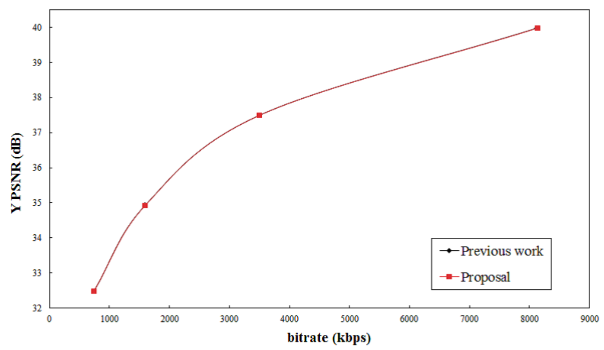

Figure 6. RD curve: ParkScene

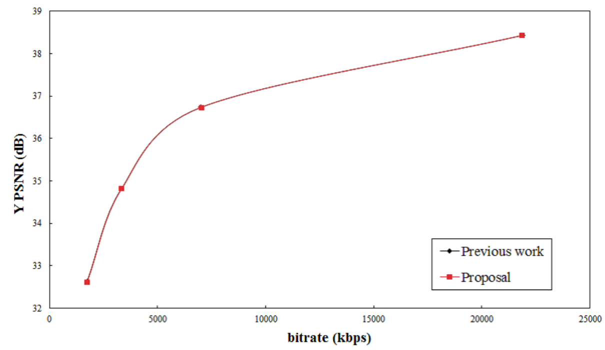

Figure 7. RD curve: Cactus

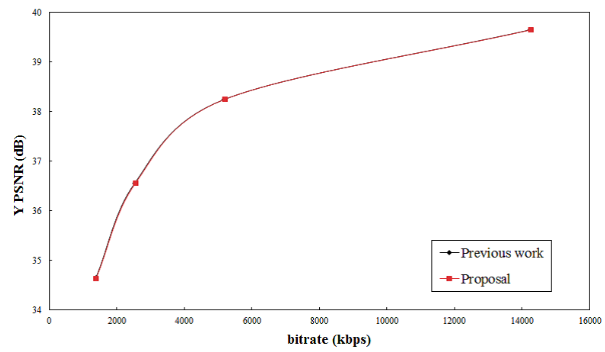

Figure 8. RD curve: BasketballDrive

4.1 Conclusion

In this paper, we proposed an adaptive search range selection algorithm to reduce computation complexity of motion

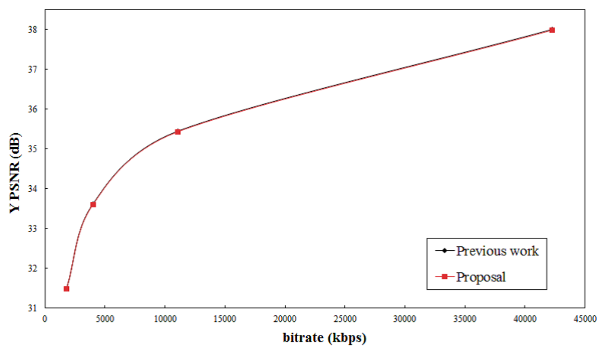

Figure 9. RD curve: BQTerrace

estimation in full search algorithm. The method consists of two step. Firstly, when the SR which is decided by variance σ^2 is redundant, our proposal modifies the SR. Secondly, our proposal predicts the distribution of the best point within the search area using MV distribution of the neighboring blocks. It reduces the redundant search area considering MV distribution. The simulation results show that the proposed algorithm achieved averagely 76.0% complexity reduction with 0.4% BD-rate increasing, compared with previous work.

References

[1] T. Wiegand, G.J. Sullivan, G. Bjntegaard, A.K. Luthra, "Overview of the H.264/AVC video coding standard," IEEE Transactions on Circuits and Systems for Video Technology, vol. 13, pp. 560-576, July 2003.

[2] R. Li, B. Zeng, M.L. Liou, "A new three-step search algorithm for block motion estimation," IEEE Transaction on Circuits and Systems for Video Technology, vol. 4, pp. 438-442, Aug. 1994.

[3] L.-M. Po, W.-C. Ma, "A novel four-step search algorithm for fast block motion estimation," IEEE Transaction on Circuits and Systems for Video Technology, vol. 6, pp. 313-317, June 1996.

[4] S. Zhu, K.-K. Ma, "A new diamond search algorithm for fast block matching motion estimation," IEEE Transaction on Image Processing, vol. 9, pp. 287-290, Feb. 2000.

[5] C.-H. Cheung, L.-M. Po, "A novel cross-diamond search algorithm for fast block motion estimation," IEEE Transaction on Circuits and Systems for Video Technology, vol. 12, pp. 1168-1177, Dec. 2002.

[6] C. Zhu, X. Lin, L.P. Chau, "Hexagon-based search pattern for fast block motion estimation," IEEE Transaction on Circuits and Systems for Video Technology, vol 12, pp. 349-355, May 2002.

[7] C.C. Lou, S.W. Lee, and C.-C. Jay Kuo, "Adaptive motion search range prediction for video encoding," IEEE Transactions on Circuits and Systems for Video Technology, vol. 20, no. 12, pp. 1903-1908, Dec. 2010.

[8] C.C. Lou, S.W. Lee, and C.-C. Jay Kuo, "Motion vector search window prediction in memory-constrained systems," Proceedings of SPIE - The International Society for Optical Engineering Applications of Digital Image Processing XXXII, vol. 7443, p. 74430E, Aug. 2009.