

A Coding Efficiency Improvement Algorithm for Future Video Coding

Xiantao Jiang¹(✉), Xiaofeng Wang¹, Yadong Yang¹, Tian Song²,
Wen Shi², and Takafumi Katayama²

¹ Department of Information Engineering,
Shanghai Maritime University, Shanghai, China
xtjiang@shmtu.edu.cn

² Department of Electrical and Electronics Engineering,
Tokushima University, Tokushima, Japan

Abstract. Motion estimation is critical in motion coding. However, the fixed pattern of search center decision method in HEVC is lack of precision. Taking advantage of the surrounding coding unit information, a universal motion vector prediction framework is presented to improve the coding efficiency in HEVC. The proposed framework is composed of two parts: motion vector prediction (MVP) and search range (SR) selection. Firstly, a novel motion vector prediction (NMVP) method is presented to improve the coding efficiency. Secondly, an adaptive search range selection (ASRS) method is developed to reduce the coding complexity in motion estimation. The simulation results demonstrate that the overall bitrate can be reduced by 5.49% on average, up to 9.18% compared with HEVC reference software.

Keywords: High efficiency video coding · Coding efficiency · Motion vector prediction · Search range

1 Introduction

In HEVC standard, the performance of motion estimation (ME) highly depended on the selection of advanced motion vector prediction (AMVP) technology [1]. The motion vector prediction (MVP) is selected from a motion vector candidate list which consists of one motion vector from neighboring units on the left of the current coding unit, one motion vector from above neighboring units, and the motion vector of the spatially the same position in the previous encoded frame. And the motion vector in the list with minimum cost is selected as the final MVP. AMVP is significantly simplified to provide a good trade-off between coding efficiency and an implementation cost. However, the fixed pattern of the MVP decision process without consideration of the reliability of the surrounding motion vectors makes it has lower estimation accuracy.

Some previous works are proposed to improving the performance of MV coding. The main idea based on the spatial and temporal MVP candidate

schemes for video coding have one assumption in common. The motion of neighboring blocks has to be similar [2–8]. Lin et al. present a new location of the temporal motion vector predictor, a priority-based derivation algorithm of spatial and temporal MVPs [1, 2]. Jung et al. propose the motion vector competition (MVC) scheme to select one motion vector predictor among the given motion vector candidates [3]. These methods can increase the coding efficiency of motion vector coding in HEVC. Yang et al. define a predictor candidate set, and the motion vector predictor can be generated by the minimum motion vector difference criterion [4]. Chien et al. design an enhanced AMVP mechanism to get an accurate motion vector predictor [5]. However, the spatial and temporal MVP candidates lack precision, and these approaches improve the performance of motion vector coding limitedly.

Motion estimation is a core part of video coding, and it can improve the coding efficiency significantly. Meanwhile, the coding complexity has significantly increased. Some adaptive search range (SR) methods have been presented to reduce the encoding complexity. Determining a suitable SR, they can be classified two categories: MV-based method and SAD-based method.

The MV-based methods [10–15] persist in that, when the distribution of the MVD is concentrated in zero with a small variance, the SR can be adjusted to the small. Lou et al. present an adaptive motion search range method to reduce the memory access bandwidth [10–12]. In this method, an applicable SR is chosen to contain the optimal MV by using a probability model. In Dai’s work [13], an adaptive SR method is proposed by using a Cauchy distribution. These methods can reduce the encoding complexity significantly.

The SAD-based methods [16, 17] set a threshold on SAD value to decide whether the video content is motion or not. However, the SAD-based methods are unreliable.

In summary, a universal motion vector prediction framework is proposed to improve the coding efficiency in this work. For the fixed pattern of MV coding in HEVC, the main disadvantages of the previous work are that the precision is not sufficient and the robustness is not high. Firstly, a novel motion vector prediction method is used to generate the optimal MV. For the fixed SR adopting in motion search processing, there are a large number of redundant computation. Thus, an adaptive search range selection method is developed to reduce the encoding complexity of ME owing to the benefit of the accurate motion vector prediction. Different from the state-of-the-art, my approach treats the MVP selection and the SR selection jointly.

2 Proposed Method

2.1 Novel Motion Vector Prediction (NMVP)

Considering the video content with strong spatial correlations, motion vector predictor of the current CU can be generated from the adjacent CUs. The novel motion vector prediction is based on the previous work [9]. Different from the fixed pattern AMVP technology, the spatial neighborhood cluster G is composed

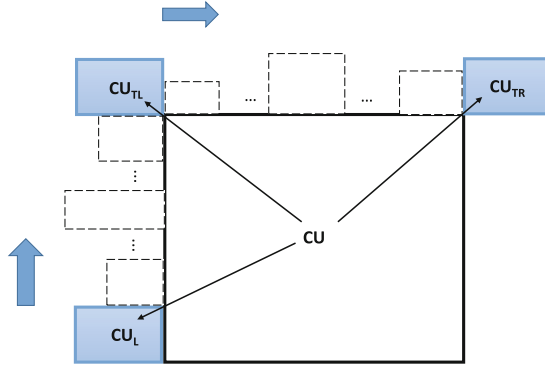


Fig. 1. Spatial correlation neighborhood cluster.

of all spatial neighbor CUs. The cluster G is shown as in Fig. 1. Where CU_L , CU_{TR} and CU_{TL} denote the left, top right and top left CU of the current CU, separately. The cluster G is defined as

$$G = \{CU_L, \dots CU_{TL}, \dots CU_{TR}\} \quad (1)$$

The MVs and depths information of G can be used to predict the MVP of the current CU. However, the computation complexity is high by the checking all of the information. Thus, the relatively reliable sub-cluster should be developed for the MVP. Therefore, in order to utilize the spatial correlation, the sub-cluster M is defined as

$$M = \{CU_L, CU_{TL}, CU_{TR}\} \quad (2)$$

The sub-cluster M is contained in the cluster G ($M \subset G$). MV_L , MV_{TR} and MV_{TL} indicate the MV candidates in the left, top right and top left of the current CU.

The basic idea of the proposed MVP method is to prejudge the MVP of the current CU according to the MVs of the spatial adjacent CUs. When the sub-cluster M is available, the information of M is used to predict the MVP of the current CU. In contrast, when the sub-cluster M is unavailable, the information of G is used to predict the MVP of the current CU.

When the MVs of sub-cluster M are available, a simple MV can be selected as the optimized MVP for the current CU. On the contrary, when the MVs of sub-cluster M are not available, the reliability of the candidate MVs is the lowest and it is hard to get the accurate MVP using the fixed AMVP mechanism. In this case, the MVP position may tend to be near to the left of CU, and it is possible to tend to be near to the top of CU. Thus, all available MVs of the spatial neighborhood cluster G need to be checked. In order to get the accurate MVP, all surrounding MVs of G can be added to the candidates MVs, and the cost of these MVs are checked to get the optimized MVP.

2.2 Adaptive Search Range Selection (ASRS)

An accurate MVP is critical to SR reduction. In this subsection, the importance of the MVP to the SR reduction is studied. As to the SR, the optimized MV is close to the MVP so that a smaller SR can be used in the ME. In the contrast, a larger SR should be used in the ME if the optimized MV is far away the MVP. The difference between the optimized MV and the MVP is named the MV prediction difference (MVPD). Thus, when the distribution of the MVPD is concentrated near the center with a small variance, a smaller SR can be adopted.

Previous works on adaptive search range selection (ASRS) methods are reported in [10, 12]. The MV variance (σ^2) of the spatial and temporal neighboring CUs is used to adjust the search range in ME. In these methods, the Cauchy distribution is proposed to model the distribution of the MVPD. The probability density function of MVPD is used to calculate the probability of the optimized MV within the SR in [10], the probability density function of zero mean Cauchy distribution is can be defined as

$$f_{ME}(x, y) = f_{ME,X}(x) \cdot f_{ME,Y}(y) \tag{3}$$

$$f_{ME,X}(x) = \frac{C_x}{|\frac{x}{\zeta_x}|^{\frac{5}{3}} + 1} \tag{4}$$

$$f_{ME,Y}(y) = \frac{C_y}{|\frac{y}{\zeta_y}|^{\frac{5}{3}} + 1} \tag{5}$$

where C_x, C_y are normalization constants, and ζ_x and ζ_y are parameters of the modified zero-mean Cauchy function, which can be computed by the sample variances of MVPDs (σ^2). Thus, the probability of the optimized MV can be defined as

$$F_{ME}(SR_x, SR_y) = F_{ME,X}(SR_x) \cdot F_{ME,Y}(SR_y) \tag{6}$$

$$F_{ME,X}(SR_x) = \int_{-SR_x-0.5}^{SR_x+0.5} f_{ME,X}(x)dx \tag{7}$$

$$F_{ME,Y}(SR_y) = \int_{-SR_y-0.5}^{SR_y+0.5} f_{ME,Y}(y)dy \tag{8}$$

where SR_x and SR_y are horizontal and vertical component of search range, separately. For the given probability C_{prob} that denotes the search range contains the optimized MV, the dynamic search range can be determined by the closest SR with a probability no less than C_{prob} , and it satisfies

$$F_{ME,X}(SR_x) = F_{ME,Y}(SR_y) = \sqrt{C_{prob}} \tag{9}$$

where the computational dynamic search range can be represented by SRx_{dyn} and SRy_{dyn} , respectively. In this work, the default SR is represented by SR_{def} . When the MVP is accurate enough, the minimal search range (represented by SRx_{min} and SRy_{min}) is set to 1/8 of the default SR.

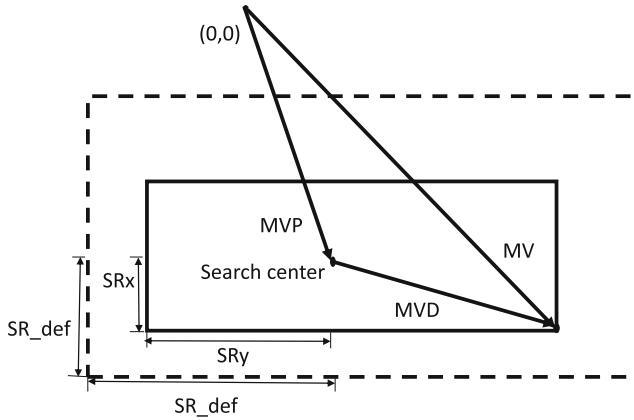


Fig. 2. Search range refinement

Recall that the spatial correlation based MVP decision algorithm is studied in above section, and the reliability of the neighboring candidate MVs is used to get the optimized MVP. To treat the MVP selection and the SR reduction jointly, when a high precision MVP is selected to be close to the optimized MV, the SR will be small. An example of the search range refinement method is shown in Fig. 2. In this case, the SR of modified search range is set to SR_x and SR_y , while the default SR (SR_def) is set to 64×64 in the ME of HEVC reference software.

2.3 Universal Motion Vector Prediction Framework

Joining the NMVP and ASRS methods together, the universal motion vector prediction algorithm is shown as algorithm 1. Firstly, the optimized MVP is decided by the MVs of the spatial neighbor CUs. Secondly, the search range is adjusted by the probability of the optimized MV within the SR and the reliability of the MV candidates. It is noted that, when the reliability of the neighboring candidate MVs is the highest, the SR can be reduced significantly and it is set to the (SR_x_min, SR_y_min) . Moreover, when the reliability of the neighboring candidate MVs is the lowest, the SR is set to the $\{\min(SR_def, SR_x_min + SR_x_dyn), \min(SR_def, SR_y_min + SR_y_dyn)\}$ respectively.

3 Experiment Results

The proposed algorithm is implemented and verified based on HEVC test model HM16.12. The test conditions are set to evaluate the performance of the proposed algorithm at different profiles (RA and LD) [18]. The quantization parameters (QP_i) are set to 22, 27, 32 and 37, respectively. In this work, the search strategy is TZsearch, and the SR is adjusted adaptively.

Algorithm 1. Universal motion vector prediction.

```

1 Start inter prediction for CU
2 if the sub-cluster  $M$  exist then
3   if  $MV_{TR} = MV_{TL} = MV_L$  then
4      $MV_L$  is selected as best MVP;
5      $SR_x = SR_{x\_min}$ ,  $SR_y = SR_{y\_min}$ ;
6   else if  $|MV_{TR} - MV_{TL}| > |MV_{TL} - MV_L|$  then
7     Add  $MV_{TL}$  and  $MV_L$  to MV candidate list;
8      $SR_x = \min(SR_{def}, SR_{x\_dyn})$ ,
9      $SR_y = \min(SR_{def}, SR_{y\_min} + SR_{y\_dyn})$ ;
10  else
11    Add  $MV_{TR}$  and  $MV_{TL}$  to MV candidate list;
12     $SR_x = \min(SR_{def}, SR_{x\_min} + SR_{x\_dyn})$ ,
13     $SR_y = \min(SR_{def}, SR_{y\_dyn})$ ;
14 else
15   Add all MVs of the cluster  $G$  to MV candidate list;
16   Reduce the redundant MV candidates;
17    $SR_x = \min(SR_{def}, SR_{x\_min} + SR_{x\_dyn})$ ,  $SR_y = \min(SR_{def},$ 
18    $SR_{y\_min} + SR_{y\_dyn})$ ;
18 Process motion search

```

The performance of the proposed algorithm is evaluated Bjontegarrd Delta bitrate (BDBR) [19], and the average time increasing (TI) is defined as

$$TI(\%) = \frac{1}{4} \sum_{i=1}^{i=4} \frac{T_{Pro}(QP_i) - T_{HM}(QP_i)}{T_{HM}(QP_i)} \times 100\% \quad (10)$$

where $T_{HM}(QP_i)$ and $T_{pro}(QP_i)$ are the encoding time by using the HEVC reference software and the proposed method with different QP_i .

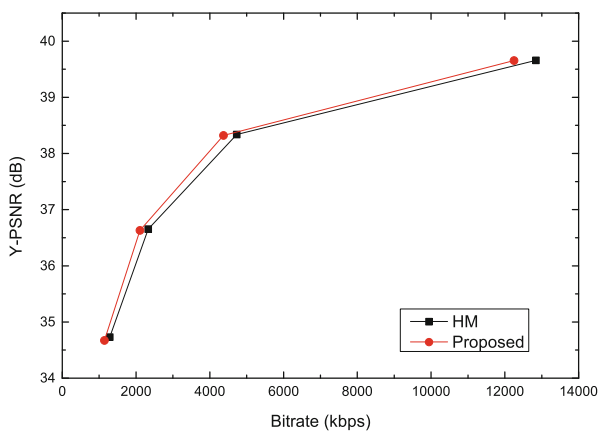
Table 1 shows the performance of the universal motion vector prediction algorithm. In RA case, the bitrate can be reduced by 5.49% on average, while the encoding time increasing is 51.13%. In LD case, the bitrate can be reduced by 5.34% on average, while the encoding time increasing is 44.96%. The proposed algorithm can improve the coding efficiency significantly.

It is noted that, for the motion severe sequence, the proposed algorithm can improve the performance, which is the greatest contribution of this paper. The bitrate can be reduced by 7.85% for BasketballDrive sequence in RA case, and the R-D curve is shown as Fig. 3.

It would be specially mentioned that this proposed method causes the encoding complexity increasing with the encoding efficiency raising. However, for the application that does not care about the real-time encoding, and care more about the coding efficiency, and it is an efficient approach for coding efficiency improvement in HEVC.

Table 1. Performance of universal motion vector prediction.

Class	Sequence	RA		LD	
		BDBR	TI (%)	BDBR	TI (%)
1920 × 1080	Kimono	-5.92	51.64	-4.46	45.41
	ParkScene	-4.35	56.79	-4.88	52.13
	Cactus	-4.76	54.68	-4.55	48.29
	BasketballDrive	-7.85	51.14	-6.04	51.03
	BQTerrace	-2.15	54.67	-2.77	50.59
1280 × 720	Vidyo1	-3.61	63.92	-5.30	60.75
	Vidyo3	-4.66	59.53	-4.29	51.20
	Vidyo4	-5.17	59.12	-5.84	52.72
High resolution	Average	-4.81	56.44	-4.77	51.52
832 × 480	BasketballDrill	-7.30	48.46	-7.13	42.56
	BQMall	-4.69	53.65	-5.10	47.36
	PartyScene	-4.60	45.68	-4.51	35.83
	RaceHorses	-8.93	41.63	-6.72	33.40
416 × 240	BasketballPass	-6.18	56.13	-6.17	34.97
	BQSquare	-2.79	54.77	-3.84	36.29
	BlowingBubbles	-5.74	47.41	-5.92	36.59
	RaceHorses	-9.18	18.89	-7.95	40.23
Low resolution	Average	-6.18	45.83	-5.92	38.40
Average		-5.49	51.13	-5.34	44.96

**Fig. 3.** R-D curve of BasketballDrive (RA).

4 Conclusion

In this work, a universal motion vector prediction framework is presented to improve the performance of HEVC. The simulation results demonstrate that the proposed overall algorithm can improve the encoding efficiency by 5.34–5.49% on average, which the encoding time increasing is about 45–51%.

References

1. Lin, J.L., Chen, Y.W., Huang, Y.W., Lei, S.M.: Motion vector coding in the HEVC standard. *IEEE J. Sel. Top. Signal Process.* **7**(6), 957–968 (2013)
2. Lin, J.L., Chen, Y.W., Tsai, Y.P., Huang, Y.W., Lei, S.M.: Motion vector coding techniques for HEVC. In: *International Workshop on Multimedia Signal Processing (MMSP)*, pp. 1–6 (2011)
3. Jung, J., Bross, B., Chen, P., Han, W.-J.: Description of core experiment 9 (CE9): MV coding and skip/merge operations. In: *Document of Joint Collaborative Team on Video Coding, JCTVC-D609*, January 2011
4. Yang, W., Au, O.C., Dai, J., Zou, F., Pang, C., Liu, Y.: Motion vector coding algorithm based on adaptive template matching. In: *International Workshop on Multimedia Signal Processing (MMSP)*, pp. 222–227 (2010)
5. Chien, W.D., Liao, K.Y., Yang, J.F.: Enhanced AMVP mechanism based adaptive motion search range decision algorithm for fast HEVC coding. In: *IEEE International Conference on Image Processing (ICIP)*, pp. 3696–3699, October 2014
6. Tok, M., Glantz, A., Krutz, A., Sikora, T.: Parametric motion vector prediction for hybrid video coding. In: *Picture Coding Symposium (PCS)*, pp. 381–384 (2012)
7. Tok, M., Eiselein, V., Sikora, T.: Motion modeling for motion vector coding in HEVC. In: *Picture Coding Symposium (PCS)*, pp. 154–158 (2015)
8. Springer, D., Simmet, F., Niederkorn, D., Kaup, A.: Robust rotational motion estimation for efficient HEVC compression of 2D and 3D navigation video sequences. In: *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 1379–1383 (2013)
9. Ikegita, J., Jiang, X., Song, T., Leu, J.-S., Shimamoto, T.: Efficient prediction motion vector candidate selection algorithm for HEVC. In: *ITC-CSCC: International Technical Conference on Circuits Systems, Computers and Communications*, pp. 402–403 (2015)
10. Lou, C.C., Lee, S.W., Kuo, C.C.J.: Motion vector search window prediction in memory-constrained systems. In: *SPIE Optical Engineering Applications. International Society for Optics and Photonics*, p. 74430E (2009)
11. Lou, C.C., Lee, S.W., Kuo, C.C.J.: Adaptive search range selection in motion estimation. In: *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 918–921 (2010)
12. Lou, C.C., Lee, S.W., Kuo, C.C.J.: Adaptive motion search range prediction for video encoding. *IEEE Trans. Circuits Syst. Video Technol.* **20**(12), 1903–1908 (2010)
13. Dai, W., Au, O.C., Li, S., Sun, L., Zou, R.: Adaptive search range algorithm based on Cauchy distribution. In: *Visual Communications and Image Processing (VCIP)*, pp. 1–5 (2012)
14. Shi, Z., Fernando, W.A.C., Fernando, A.: Adaptive direction search algorithms based on motion correlation for block motion estimation. *IEEE Trans. Consum. Electron.* **57**(3), 1354–1361 (2011)

15. Ko, Y.H., Kang, H.S., Lee, S.W.: Adaptive search range motion estimation using neighboring motion vector differences. *IEEE Trans. Consum. Electron.* **57**(2), 726–730 (2011)
16. Lee, S.W., Park, S.M., Kang, H.S.: Fast motion estimation with adaptive search range adjustment. *Opt. Eng.* **46**(4), 040504–0405043 (2007)
17. Saponara, S., Casula, M., Fanucci, L., Rovati, F., Alfonso, D.: Dynamic control of motion estimation search parameters for low complex H. 264/AVC video coding. In: *IEEE International Conference on Consumer Electronics*, pp. 481–482 (2006)
18. Bossen, F.: Common test conditions and software reference configurations. In: *JCTVC-L1100*, January 2013
19. Bjontegaard, G.: Calculation of average PSNR differences between RD-curves. In: *ITU-T SG16 Q.6, VCEG-M33*, April 2001