History-based Motion Vector Prediction in Versatile Video Coding

Li Zhang[*], Kai Zhang[*], Hongbin Liu[+], Hsiao Chiang Chuang[*], Yue Wang[+], Jizheng Xu[*], Pengwei Zhao[+], Dingkun Hong[+]

[*]*Bytedance Inc.*
8910 University Center Ln
San Diego, CA, USA

[+]*Beijing Bytedance Network Technology Co., Ltd*
56 Zhichun Rd, Haidian District
Beijing, China

{lizhang.idm, zhangkai.video, liuhongbin.01, chuangh, wangyue.v, xujizheng, zhaopengwei, hongdingkun}@bytedance.com

***Abstract**:* In this paper, History-based Motion Vector Prediction (HMVP) is presented for video coding. With the proposed method, a table of HMVP candidates is maintained and updated on-the-fly. After decoding one inter-coded block, the table is updated by appending the associated motion information to the table as a new HMVP candidate. A First-In-First-Out (FIFO) rule is applied to manage the table. The HMVP candidates could be added to the Advanced Motion Vector Prediction (AMVP) candidate list as additional motion vector predictors. And they could also be added to the merge candidate list as additional merge candidates. With the proposed method, the motion information of previously coded blocks even not adjacent to the current block can be utilized for more efficient motion vector prediction. Simulation results have validated the efficiency of HMVP, wherein up to 4% BD rate saving could be achieved. The proposed method has been adopted by the next generation video coding standard, named Versatile Video Coding (VVC) developed by Joint Video Exploration Team (JVET).

1. Introduction

The state-of-art video coding standard, known as High Efficiency Video Coding (HEVC), developed by Joint Collaborative Team on Video Coding (JCT-VC) with experts from ITU-T Video Coding Experts Group (VCEG) and the ISO/IEC Moving Picture Experts Group (MPEG), has been finalized in Jan. 2013 [1]. Compared to its predecessor H.264/AVC, HEVC could provide approximately 50% BD rate savings for equivalent perceptual quality. However, there are still increasing industrial requirements to further reduce the video transmission bandwidth and enhance the quality of user experience [2]. To study the technical possibilities for a next generation video codec, VCEG and MPEG have been working together on this exploration in a joint collaboration team known as the Joint Video Exploration Team (JVET) to evaluate new compression techniques beyond HEVC. JVET was founded in Oct. 2015, and after one and half years, the reference software for the JVET group named Joint Exploration Model (JEM) can achieve around 28% BD rate reduction compared to HEVC under the Random Access (RA) configurations. This result provides a sufficient evidence that there is good potential to develop a new generation video codec. Therefore, the next generation video coding standard, named Versatile Video Coding (VVC), has been launched from Apr. 2018 which is still under JVET [3].

2375-0359/19/$31.00 ©2019 IEEE
DOI 10.1109/DCC.2019.00012

Inter-prediction is crucial to most video coding standards, such as H.265/HEVC and VVC. Generally speaking, inter-prediction utilizes motion information to derive a motion compensated prediction (MCP) block. The difference between the MCP block and the original block instead of the original block itself is transformed, quantized and entropy coded, such that the temporal redundancies could be greatly reduced.

Although inter-prediction technologies have been studied for many years, there are still several aspects worth to be investigated to further improve its efficiency. Firstly, to handle the scenarios with significant temporal illumination variations, localized weighted prediction [4] is proposed to employ block-level linear regression improvement model. To address this issue in an alternative way, generalized bi-prediction (GBi) technique which extends the notion of the existing weighted bi-prediction to block level presented in [5]. Secondly, to capture non-translational motion information, more sophisticated motion models are proposed, such as the affine motion model [6][7]. Thirdly, to better code the motion information, a bundle of approaches have been invented. In HEVC and VVC, there are two ways to code the motion information: merge mode and advanced motion vector prediction (AMVP) mode. They both try to predict the motion information of current block by exploiting spatial and temporal correlations between the current block and neighboring blocks. With AMVP mode, the Motion Vector Difference (MVD) is further signaled. While with the merge mode, only one index of motion candidates is signaled, assuming the MVD to be zero.

Several methods have been developed to further improve the merge mode and the AMVP mode. In one direction, sub-block merge candidates are introduced, wherein one block is split to multiple sub-blocks and each sub-block is associated with its own motion information [8]. As a second direction, template matching is applied to re-estimate the applicable motion parameters for the merge mode at the decoder side [9][10]. In yet another direction, the coding of MVDs in the AMVP mode could also be improved. Girod [11] first gives a theoretical analysis for the coding gain by increasing the MV resolution. A frame level adaptive motion vector resolution selection scheme based on a rate-distortion model is then presented in [12].

In this paper, History-based Motion Vector Prediction (HMVP) is presented. Different from those prior-arts which are only beneficial for one inter coding mode, the coding performance of both AMVP and merge modes could be improved with HMVP. More specifically, motion information of previously coded blocks which may be far away from current block, can be treated as HMVP candidates. An HMVP candidate can be utilized in the AMVP or merge candidate list construction processes. As a low complexity design, a table with a limited number of HMVP candidates is updated on-the-fly in the coding order. Only the latest several HMVP candidates are stored in the table. When coding a new inter-coded block, the HMVP candidate is used as a motion vector predictor candidate. After coding an inter-coded block, the table is updated with the newly coded motion information. With the proposed HMVP method, over 1% BD rate reduction can be achieved under the RA and Low Delay P (LDP) configurations.

The rest of this paper is organised as follows. In Section 2, we review the major features of inter-prediction in HEVC. Then the proposed history-based motion vector prediction algorithm is described in Section 3. Simulation results are presented and discussed in Section 4. Finally, conclusions are drawn in Section 5.

2. Review of inter prediction

In HEVC, motion information of inter-coded blocks can be signaled in two ways: the AMVP mode and the merge mode. For either of the two coding modes, a motion vector prediction (MVP) candidate list needs to be constructed. With the AMVP mode, a reference index, a MVP candidate index referring to an AMVP candidate list and a difference between the true motion vector and the selected MVP candidate is signaled. While with the merge mode, only a merge index referring to a merge candidate list is signaled and all the motion information associated with the merge candidate is inherited. The details of the candidate list construction process of the AMVP mode and merge mode are described in the following sub-sections.

2.1. AMVP mode

AMVP exploits spatial-temporal correlation of motion vector with neighbouring blocks, which is used for explicit transmission of motion parameters. For each reference picture list, a motion vector candidate list is constructed by firstly checking availability of left, above temporally neighbouring positions, removing redundant candidates and adding zero vector to make the candidate list to be constant length. Then, the encoder select the best predictor from the candidate list and transmit the corresponding index indicating the chosen candidate.

For spatial motion vector candidate derivation, two motion vector candidates are eventually derived based on motion vectors of blocks located in five different positions as depicted in Figure . The five neighboring blocks located at B_0, B_1, B_2, and A_0 A_1 are classified into two groups, Group A includes the three above spatial neighboring blocks and Group B includes the two left spatial neighbouring blocks. The two MV candidates are derived from Group A and Group B respectively and added to the AMVP candidate list.

For temporal motion vector candidate derivation, one motion vector candidate is derived based on two different co-located positions checked in order, bottom-right and central of the co-located block, C0 and C1 in a reference picture as depicted in Figure . To avoid redundant MV candidates, duplicated motion vector candidates in the list are removed. If the number of potential candidates is smaller than two, additional zero motion vector candidates are added to the list.

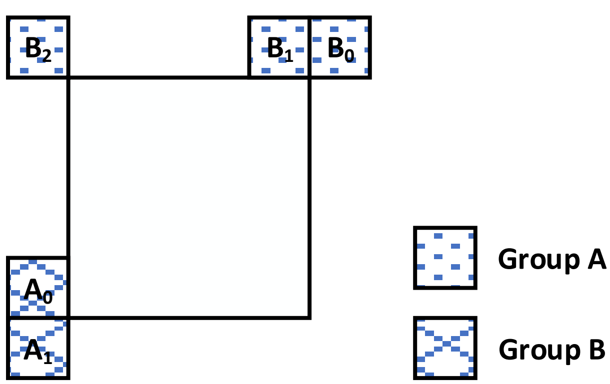


Figure 1. Positions of spatial neighbouring blocks used in AMVP/merge candidate list construction

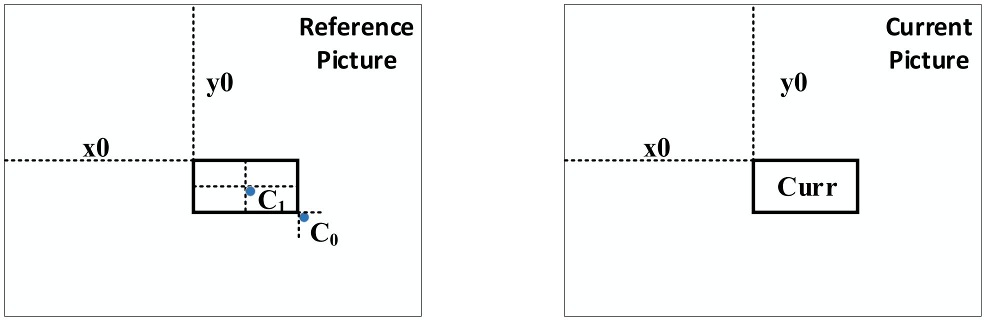


Figure 2. Temporal candidate positions, C0 and C1.

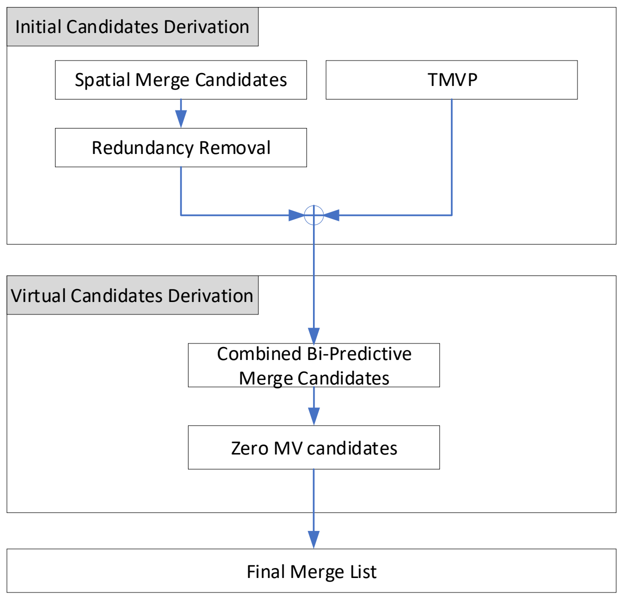


Figure 3. Merge canddiate list construction process.

2.2. Merge mode

Similar to the AMVP candidate list construction process, the merge candidate list is also derived from temporal and spatial neighboring coded blocks. The detailed sequential steps of the merge list construction process are depicted in Figure . Basically, there are two stages: initial candidate derivation from neighboring coded blocks and virtual candidate derivation to fill in the remaining merge list.

For the initial candidate derivation, it further includes spatial candidate derivation process, redundancy check for spatial candidates and temporal candidate derivation process. In the spatial merge candidate derivation, several candidates derived from five different spatial neighboring blocks may be added to the merge list after redundancy check. The five spatial neighboring blocks are the same as those used in the AMVP process. For temporal merge candidate (TMVP) derivation, two temporal neighboring blocks may be utilized to derive the TMVP candidate. The two temporal neighbouring blocks are the blocks located at bottom-right and center of the co-located block in the co-located picture. The two positions are denoted by C_0 and C_1 respectively as shown in Figure . If the block at position C_0 is not available, for example, it is intra coded, or is outside of the current CTU row, position C_1 is used. Otherwise, position C_0 is used in the derivation of the temporal merge candidate. When there are not enough merge candidates from the initial candidate derivation process, additional candidates are generated. Once the number of available merge candidates reaches the signaled maximally allowed merge candidates, the merge candidate list construction process is terminated.

3. History-based motion vector prediction

In this section, the motivation of HMVP is firstly discussed. Then, the coding flow of the proposed HMVP method is introduced, followed by how the HMVP candidates are utilized to further improve the inter-prediction efficiency.

3.1. Motivation of HMVP

As described in Section 2, spatial-temporal correlation of motion information with neighbouring blocks is utilized in the AMVP and merge candidate list construction process. Although there is a high possibility that the current block and its neighbouring blocks belong to the same object and with the same motion trajectory, it is possible that one block may be more correlated with a non-adjacent block in some scenarios such as object occlusion. For yet another reason, when there are not enough motion candidates available from the spatial/temporal neighboring blocks, virtual motion candidates are generated to fill in the candidate list. The chances of choosing the virtual motion candidates are relative low. Motion information gathered from coded blocks far from the current block may be more efficient than those virtual ones.

3.2. Coding flow of HMVP

To solve the problems mentioned above, one straightforward way is to fetch motion information from non-adjacent blocks. However, it requires significant complexity overhead in terms of memory access and line buffer sizes, especially for hardware implementation. A smarter way is to create a limited buffer to store the previously coded motion information. The motion information of a previously coded block is defined as an HMVP candidate. Multiple HMVP candidates are stored in a table, named as the HMVP table, and this table is maintained during the encoding/decoding process on-the-fly. The HMVP table is emptied when starting coding/decoding a new slice. Whenever there is an inter-coded block, the associated motion information is added to the last entry of the table as a new HMVP candidate. The overall coding flow is depicted in Figure.

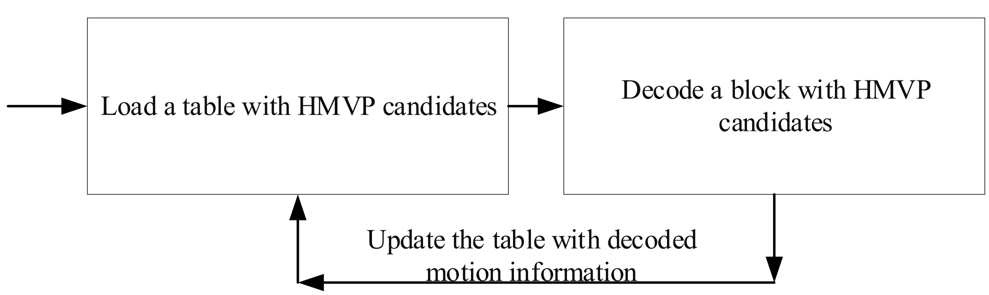


Figure 4. Decoding flow chart with the proposed HMVP method.

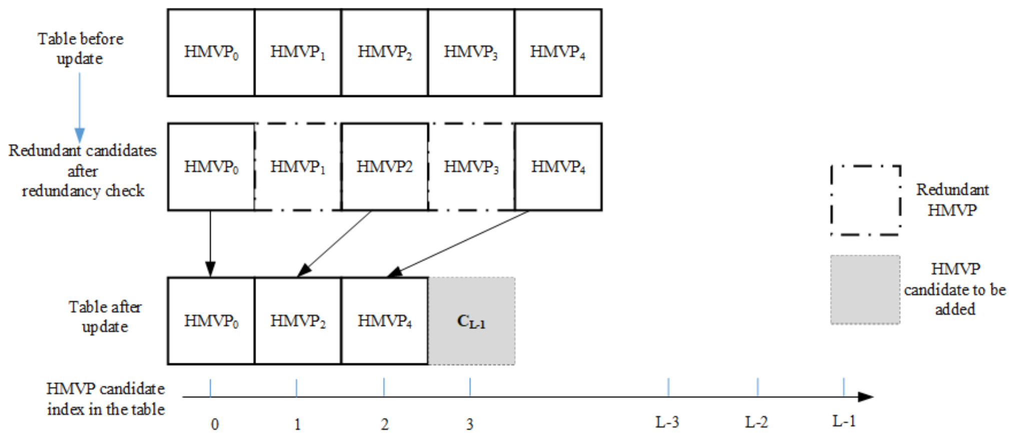


Figure 5. Example of table updating process.

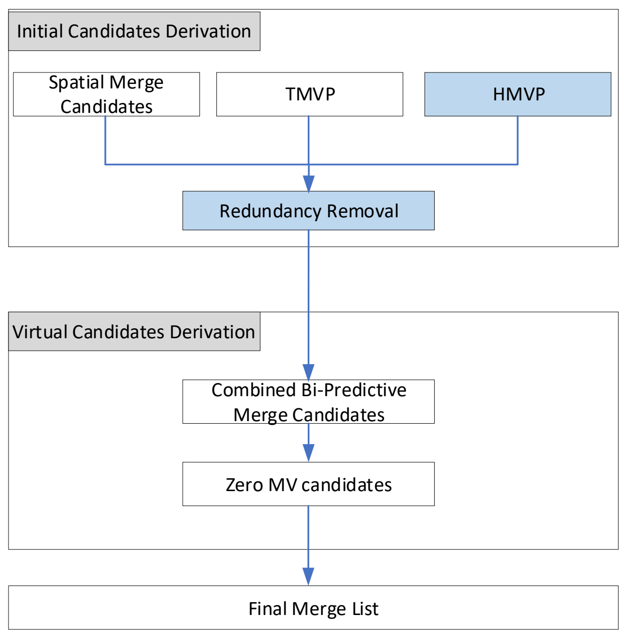


Figure 6. Modified merge list construction process.

In this paper, the HMVP table size is set to be 16, which indicates up to 16 HMVP candidates can be added to the table. If there are more than 16 HMVP candidates from the previously coded blocks, a First-In-First-Out (FIFO) rule is applied so the table always contains the latest 16 motion candidates previously coded. When appending a HMVP to the table, redundancy check is firstly conducted to find whether there is an identical HMVP in the table. If found, the identical HMVP is removed from the table and all the HMVP candidates behind it are moved forward with indices reduced by 1. If no redundancy is found and the HMVP table size is 16, the first HMVP candidate is removed and all other HMVP candidates are moved forward. Finally, the new HMVP candidate is appended at the end of the table.

Figure depicts an example wherein the FIFO rule is applied to remove a HMVP candidate and add a new one to the table. Suppose there are L HMVP candidates in the table denoted by $HMVP_i$, wherein i denotes the HMVP candidate index and i is within the range of [0, L-1], as shown in the first row of Figure . When a new HMVP candidate denoted as C_{L-1} is to be added, it is firstly compared to all existing HMVP candidates. If C_{L-1} is identical to $HMVP_2$, all the HMVP candidates with indices larger than 2 will be firstly moved forward toward the head of the table. Finally, C_{L-1} is put at the end of the HMVP table.

3.3. Usage of HMVP to the merge mode

HMVP candidates could be used in the merge candidate list construction process. The modified merge candidate list construction process is depicted in Figure 6. When the merge candidate list is not full after the TMVP candidate insertion, HMVP candidates stored in the HMVP table could be utilized to fill in the merge candidate list. Considering that one block usually has a higher correlation with the nearest neighbouring block in terms of motion information, the HMVP candidates in the table are inserted in a descending order of indices. The last entry in the table is firstly added to the list, while the first entry is added in the end. Similarly, redundancy removal is applied on the HMVP candidates. Once the total number of available merge candidates reaches the maximal number of merge candidates allowed to be signaled, the merge candidate list construction process is terminated.

3.4. Usage of HMVP to the AMVP mode

Similarly, HMVP candidates could also be used in the AMVP candidate list construction process. The motion vectors of the latest several HMVP candidates in the HMVP table are inserted after the TMVP candidate. For each HMVP candidate, if it has the same reference index as the AMVP target reference index, the corresponding motion vector is added to the AMVP candidate list. Redundancy check is applied on the HMVP candidates to remove identical MVs. The total size of the AMVP candidate list is kept to be two as in HEVC.

4. Simulation results and complexity analysis

To demonstrate the coding performance of the proposed method, two sets of simulations have been conducted. In the first test set, the coding gain of HMVP over VTM version 1.0 (VTM-1.0) [13] is evaluated with the same merge list size. In the second test set, the coding gain of HVMP is further tested with the increased merge list size to 15. All experiments are conducted following the common test conditions in JVET activities [14].

Software used in this experiment is based on the recent VTM-1.0 and the proposed HMVP method has been integrated to this platform. Since the proposed method is applied to inter-prediction, three configurations used in the JVET were tested, i.e., random access Main 10, low delay P Main 10 and low delay B Main 10. Quantization Parameters (QPs) used in the tests are 22, 27, 32, and 37. The coding efficiency is measured by the BD rate that is widely used in JVET [15]. The BD rate indicates the BD rate increasement ratio over the anchor by the test when maintaining equivalent PSNR. For example, a negative value corresponding to BD rate reduction means that coding performance is improved.

Table 1 summarizes the results of the first test set. In this evaluation, the merge list size is set to be six in both the anchor and tested method. As shown in Table 1, the average BD rate reductions of HMVP are 0.8%, 0.8%, 0.3% for RA, LDP and LDB, respectively. It should be noted that for this test set, the encoder complexity is almost the same as the anchor since the number of full rate-distortion checks are kept unchanged. Therefore, the BD rate saving shown in this set purely demonstrates the efficiency of HMVP candidates. In addition, to further check the potential gain of HMVP, Table 2 presents the results of the second test set. In this evaluation, the merge list size is increased to 15 even we keep the number of full rate-distortion checks unchanged. As shown in this table, the average coding gains of HMVP are even higher, i.e., 1.1%, 1.2%, 0.6% for RA, LDP and LDB, respectively. Moreover, Figure shows the detailed results for each sequence in the RA and LDP configuration. The largest gain comes from the sequence “BQTerrance”, with 3% and 4% BD rate savings for RA and LDP, respectively.

Table 1. Coding gain of HMVP over VTM-1.0 (with same merge list size)

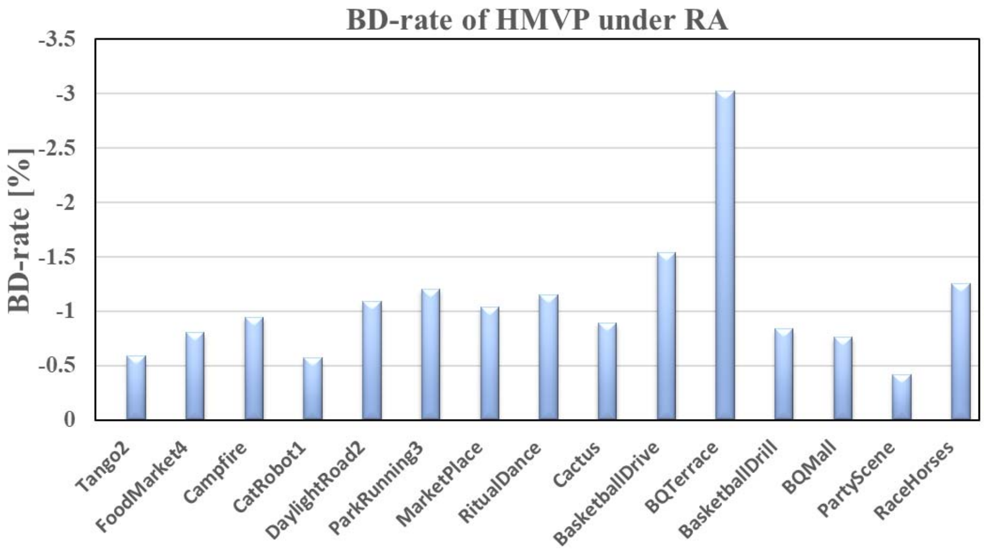
Class	Random Access Main 10					Low Delay P Main 10					Low Delay B Main 10				
	Y	U	V	EncT	DecT	Y	U	V	EncT	DecT	Y	U	V	EncT	DecT
Class A1	-0.6%	-0.7%	-0.7%	99%	100%										
Class A2	-0.7%	-1.0%	-1.0%	99%	101%										
Class B	-1.1%	-1.2%	-1.0%	100%	98%	-1.1%	-0.4%	-0.4%	102%	100%	-0.6%	-0.4%	-0.2%	100%	101%
Class C	-0.7%	-0.9%	-1.0%	100%	99%	-0.7%	-0.5%	-0.7%	101%	97%	-0.3%	-0.2%	0.0%	100%	97%
Class E						-0.4%	-0.6%	-0.8%	100%	97%	-0.3%	-0.4%	-0.3%	99%	98%
Overall	-0.8%	-1.0%	-0.9%	100%	99%	-0.8%	-0.5%	-0.6%	101%	98%	-0.4%	-0.3%	-0.2%	100%	99%

Table 2. Coding gain of HMVP over VTM-1.0 (with increased merge list size)

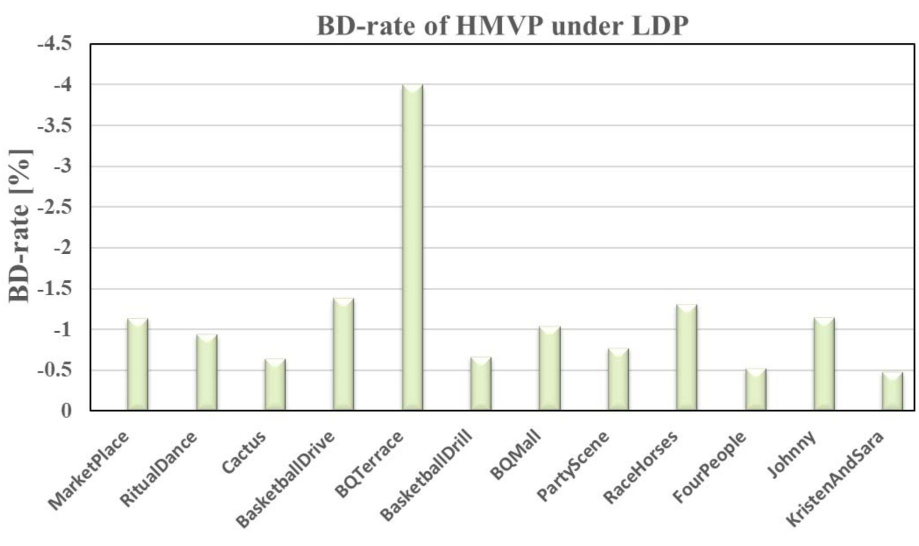
Class	Random Access Main 10					Low Delay P Main 10					Low Delay B Main 10				
	Y	U	V	EncT	DecT	Y	U	V	EncT	DecT	Y	U	V	EncT	DecT
Class A1	-0.8%	-1.0%	-1.0%	105%	101%										
Class A2	-1.0%	-1.4%	-1.3%	105%	102%										
Class B	-1.5%	-1.5%	-1.3%	106%	102%	-1.6%	-1.1%	-1.0%	110%	102%	-1.0%	-0.5%	-0.5%	108%	101%
Class C	-0.8%	-1.2%	-1.1%	105%	102%	-0.9%	-0.9%	-1.1%	109%	102%	-0.3%	-0.1%	-0.2%	109%	102%
Class E						-0.7%	-1.0%	-0.6%	110%	101%	-0.5%	-0.2%	-0.5%	107%	102%
Overall	-1.1%	-1.3%	-1.2%	105%	101%	-1.2%	-1.0%	-0.9%	110%	101%	-0.6%	-0.3%	-0.4%	107%	102%

5. Conclusions

In this paper, a history-based motion vector prediction method is proposed to code motion information more efficiently. In the proposed method, a small table is maintained at both encoder and decoder to store the latest several sets of motion information from previously coded blocks. When coding a new block, the stored motion information named as HMVP candidate may be utilized for either motion vector prediction in the AMVP mode or motion vector inheritance in the merge mode. Simulations under common test conditions defined by JVET for VVC verified the remarkable performance of the proposed method by contributing more than 1% coding gains with almost no encoding/decoding complexity burden. HMVP has been adopted to the most recent version of VVC standard.



(a) Under RA configuration



(b) Under LDP configuration

Figure 7. BD rate gains of HMVP for each test sequence

References

[1] High Efficiency Video Coding (HEVC) Standard," IEEE Trans. Circuits and Systems for Video Technology, Vol. 22, No. 12, pp. 1649-1668, Dec. 2012.

[2] ISO/IEC JTC1/SC29/WG11, "Proposed Revised Requirements for a Future Video Coding Standard," MPEG doc.M36183, Warsaw, Poland, Jun. 2015.

[3] J.-R. Ohm, G. J. Sullivan, "Versatile Video Coding – towards the next generation of video compression," Picture Coding Symposium 2018, San Francisco, June. 2018.

[4] N. Zhang, Y. Lu, X. Fan, R. Xiong, D. Zhao, W. Gao, "Enhanced inter prediction with localized weighted prediction in HEVC", 2015 Visual Communications and Image Processing (VCIP), NYU, Singapore, Dec. 2015.

[5] C.-C. Chen, X. Xiu, Y. He, Y. Ye, "Generalized bi-prediction method for future video coding," 2016 Picture Coding Symposium (PCS), 4-7 Dec. 2016.

[6] L. Li, H. Li, D. Liu, Z. Li, H. Yang, S. Lin, H. Chen, F. Wu, "An Efficient Four-Parameter Affine Motion Model for Video Coding," IEEE Trans. Circuits and Systems for Video Technology, Vol. 28, No. 8, pp. 1934-1948, 2018.

[7] K. Zhang, Y.-W Chen, L. Zhang, W.-J Chien, M. Karczewicz, "An Improved Framework of Affine Motion Compensation in Video Coding," IEEE Trans. Image Process. 2018 Oct 22. doi: 10.1109/TIP.2018.2877355 (To appear)

[8] W.-J. Chien, Y. Chen, J. Chen, L. Zhang, M. Karczewicz, X, Li, "Sub-block motion derivation for merge mode in HEVC," Proc. SPIE 9971, Applications of Digital Image Processing XXXIX, Sep. 2016.

[9] S. Kamp, M. Wien, "Decoder-Side Motion Vector Derivation for Block-Based Video Coding," IEEE Transactions on Circuits and Systems for Video Technology, Vol. 22, No. 12, pp. 1732-1745, Dec. 2012.

[10] X. Li, J. Chen, M. Karczewicz, "Frame Rate Up-Conversion Based Motion Vector Derivation for Hybrid Video Coding," Data Compression Conference (DCC), pp. 390-399, 2017.

[11] B. Girod, "Motion-compensating prediction with fractional-pel accuracy," IEEE Transactions on Communications, vol. 41, no. 4, pp. 604-612, 1993.

[12] Z. Wang, J. Zhang, N. Zhang, S. Ma, "Adaptive Motion Vector Resolution Scheme for Enhanced Video Coding," 2016 Data Compression Conference, Vol. 1, pp. 101-110, Apr. 2016.

[13] VTM 1.0: https://jvet.hhi.fraunhofer.de/svn/svn_VVCSoftware_VTM/tags/VTM-1.0/

[14] "JVET common test conditions and software reference configurations", J. Boyce, K. Suehring, X. Li, V. Seregin, doc. JVET-J1010, April 2018.

[15] G. Bjontegaard, "Calculation of average psnr differences between rd-curves," ITU-T SG.16 Q.6, Document VCEG-M33, Apr. 2001.