Simplified Triangular Partitioning Mode in Versatile Video Coding

Dohyeon PARK†, Jinho LEE††, Jung-Won KANG††, Nonmembers, and Jae-Gon KIM†a), Member

SUMMARY The emerging Versatile Video Coding (VVC) standard currently adopts Triangular Partitioning Mode (TPM) to make more flexible inter prediction. Due to the motion search and motion storage for TPM, the complexity of the encoder and decoder is significantly increased. This letter proposes two simplifications of TPM for reducing the complexity of the current design. One simplification is to reduce the number of combinations of motion vectors for both partitions to be checked. The method gives 4% encoding time decrease with negligible BD-rate loss. Another one is to remove the reference picture remapping process in the motion vector storage of TPM. It reduces the complexity of the encoder and decoder without a BD-rate change for the random-access configuration.

key words: Video Codec, Versatile Video Coding, inter prediction, triangular partitioning mode

1. Introduction

Joint Video Experts Team (JVET) founded by ISO/IEC Moving Picture Experts Group (MPEG) and ITU-T Video Coding Experts Group (VCEG) is developing a new video coding standard named as Versatile Video Coding (VVC) beyond High-Efficiency Video Coding (HEVC) [1]. JVET has been releasing Working Draft (WD) and the reference software called VVC Test Model (VTM) with new adopted coding tools in each meeting [2], [3].

In video coding, the performance of inter prediction has been improved by allowing variously shaped prediction block, but the prediction blocks still have the limit of being rectangular. In the VTM, Triangular Partitioning Mode (TPM) has been adopted for more flexible inter prediction. TPM divides a rectangular coding block (CB) into two triangular prediction blocks (TPBs) with diagonal or inverse diagonal direction as shown in Fig. 1. Each TPB has its own uni-predictive motion vector (MV) which is derived from the merge candidate list constructed by MVs of adjacent CBs. To indicate TPM operation, the index of the determined combination of partition direction and MV for each partition is signaled by using a Look-Up Table (LUT) in which all candidates are specified up to 40 cases.

According to the JVET tool evaluation report, TPM gives meaningful BD-rate saving of 0.37% [4]. However, there is a 10–12% encoding time increase, which is mainly due to the computation of rate-distortion (R-D) cost for the 40 candidates of TPM. To address such complexity issue of TPM, several contributions have been proposed and reviewed positively in JVET [5]. In the latest adoption related to TPM, the regular merge candidate list is re-used to solve the issue of using its own process for construction of merge candidate, which is different from the existing merge candidate construction. This adoption unified and simplified the merge candidate list construction, but the encoder and decoder complexities are not practically reduced. In other words, there was no runtime reduction [6].

In this letter, we propose two simplification methods for TPM to reduce both encoding and decoding complexity while keeping coding performance as much as possible [7]. The rest of the letter is organized as follows. Background section presents the details on TPM processes that cause complexity increase. The proposed simplification methods are presented in the proposed method section. After the performance analysis of the proposed methods in the experimental result section, this letter is concluded.

2. Background

In the current design of TPM, two syntaxes are defined: merge_triangle_flag and merge_triangle_idx. Basically, two syntaxes are allowed under merge mode. The merge_triangle_flag indicates whether TPM is applied or not to the current CB. Then, a merge index for each TPB and triangle partition direction are jointly signalled by merge_triangle_idx when merge_triangle_flag is true. More specifically, 40 candidates, which are specified by combining the merge indexes of the first TPB (5 cases) and the second TPB (4 cases except the selected one in the first TPB)
and the partition direction, are defined in a LUT with the size of 40 columns by 3 rows (40x3). Thus, a particular prediction of TPM can be performed among 40 candidates combining the three parameters. The selected candidate is signalled by merge_triangle_idx which ranges from 0 to 39, and its value is binarized as Exponential-Golomb code as shown in Table 1. The encoding complexity is significantly increased because the encoder should determine the best one among all candidates by comparing R-D costs. In addition, there is an additional memory burden on both the encoder and decoder to store the LUT.

In VTM, MVs of the current CB are stored in the unit of the 4x4 grid for reuse in the CBs to be coded later. In other words, either uni-predictive or bi-predictive MV of the corresponding CB can be stored in the MV storage. Likewise, for TPM, an uni-predictive MV is stored in the grids of the corresponding TPB. It can be achieved with 16 comparisons to store the motion in a corresponding direction. However, at the 4x4 grid lies on the boundary area of triangular partitions, in which both predictions are weighted combined as final prediction, a bi-predictive MV which is combined by each TPB’s MV is stored as shown in Fig. 2.

In the combination of a bi-predictive MV, two ways to combine two MVs are adaptively allowed depending on the reference pictures. If the reference pictures of the MVs are different, the 12 comparisons are required to store the motion information for the correct motion direction. Otherwise, the reference picture remapping process is performed to alternate the direction of one MV when two TPBs’ MVs have the same reference picture list. Assuming that there are two reference pictures in each reference picture list, up to 24 comparisons are required to check MVs’ reference picture lists in the reference picture remapping process of TPM. Such comparison is not required in the storage of the regular CBs’ MVs. Also, additional 28 comparisons are required to store the remapped MV. This large number of comparisons (up to 52) for specific CBs in the MV storage may cause throughput problems. The number of these comparisons can be derived straightforwardly based on the description in the section of ‘Motion vector storing process for triangle merge mode’ in the VVC working draft 3 [2].

### 3. Proposed Method

Two simplification methods for the TPM are proposed in this letter. One simplification is to reduce the number of prediction candidates for both triangular partitions in the encoder side. In the current TPM, the combinations of the split direction and the MVs of two partitions are defined up to 40 cases. Figure 3 shows the cumulative probability distribution for the occurrence of merge_triangle_idx. It is noted that indexes with values less than 16 have an occurrence probability over 80%. As a result of the empirical observations, we have found that some cases of the combinations are rarely selected. Based on this observation, the proposed method restricts the number of the combinations to be checked to 16 according to the hit occurrence of combinations. Accordingly, the size of the LUT defining combinations is reduced to 16x3 from 40x3. Besides, more efficient binarization and entropy coding are devised for the reduced index range. In other words, the binarization of the combination index is modified with fixed length coding and truncated binary coding as shown in Table 2. In the entropy coding, the first bin of the index is coded in regular mode, and the remaining bins are bypass coded for faster index coding.

Another proposed method is related to the simplification of MV storage of triangular partitions. Each TPB has a uni-predictive MV, which can be stored on the allocated 4x4

---

**Table 1** Codewords for merge_triangle_idx in VVC [2]

| Codeword form | Range of index |
|---------------|----------------|
| 0 b0          | 0 ~ 1          |
| 1 0 b1 b0     | 2 ~ 5          |
| 1 1 0 b2 b1 b0| 6 ~ 13         |
| 1 1 1 0 b3 b2 b1 b0 | 14 ~ 29 |
| 1 1 1 1 0 b4 b3 b2 b1 b0 | 30 ~ 39 |

**Fig. 2** An example of motion vector storage for the 4x4 blocks in the TPBs

**Table 2** Codewords for merge_triangle_idx in the proposed method

| Codeword form | Range of index |
|---------------|----------------|
| 0 b1 b0       | 0 ~ 3          |
| 1 0 b1 b0     | 4 ~ 7          |
| 1 1 b2 b1 b0  | 8 ~ 15         |

**Fig. 3** Cumulative distribution of merge_triangle_idx
grids in the MV storage. But, a bi-predictive MV is stored for 4x4 grids located on the boundary areas of partitions. If two partitions’ MVs have the same reference picture list, the reference picture remapping process, which converts the direction of one MV to the other one, is performed to derive a bi-predictive MV. Otherwise, a bi-predictive MV which consists of both TPBs’ MVs with different reference picture lists is stored without reference picture remapping. In the proposed method, the reference picture remapping process is removed, and the first TPB’s MV is stored for the boundary area when two partitions have the same direction. If the directions of the two MVs are different, a combined bi-predictive MV is stored as the existing method. In other words, the process that requires 52 comparisons to store the TPM motion information is pruned. As a result, the number of comparisons required to store a MV is reduced up to 16 from 52, which results in enhancement of throughput in the MV storage of TPM.

4. Experimental Results

The proposed methods are implemented on top of VTM-3.0. It is evaluated with Random Access (RA) and Low-Delay B (LDB) configurations and the JVET Common Test Condition (CTC) [8]. Each numerical result of the Bjøntegaard Delta (BD)-rate is the average results of the class-specific sequences for Quantization Parameters (QPs) of 22, 27, 32 and 37. All experimental results are compared over the anchor of VTM-3.0. Table 3 and Table 4 show the performance for the reduced number of triangular prediction candidates with the modified entropy coding. This simplification gives 4% and 6% encoding time reductions with minor BD-rate loss of 0.01% and 0.03% in RA and LDB for the luma component, respectively. i.e., the complexity of TPM is reduced to almost half. When the existing binarization of the combination index is used, noticeable coding losses of 0.05% and 0.11% in RA and LDB, respectively, are observed in the proposed simplification. Therefore, normative changes of the proposed entropy coding are essential for the coding performance.

As shown in Table 5 and Table 6, another simplification of removing reference picture remapping process achieves 0.0% and 0.17% BD-rate losses without encoding and decoding time change in RA and LDB, respectively. It is noted that hardware throughput enhancement is not directly reflected in the software runtime performance. However, the throughput issue is critical to hardware implementation and should be addressed.

5. Conclusions

This letter presents two simplification methods of the triangular partitioning mode in the emerging standard of VVC. One simplification is the reduction of the number of TPM prediction candidates. It has the minor BD-rate loss of 0.01% with the significant encoding time reduction of 4% in RA. Another simplification reduces the complexity of encoder and decoder by removing the reference picture remapping process without BD-rate loss for RA. In addition, both simplifications are disjoint and can be applied in a combined way without performance redundancy.

Acknowledgments

This work was partly supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No. 2016-0-00572 and 2017-0-00486).
References

[1] High Efficiency Video Coding, Version 1, Rec. ITU-T H.265, ISO/IEC 23008-2, Jan. 2013.
[2] B. Bross, J. Chen, and S. Liu, “Versatile video coding (draft 3),” 12th JVET Meeting, Doc. JVET-L1001, Oct. 2018.
[3] VVC Test Model (VTM), https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM/
[4] W. Chien and J. Boyce, “JVET AHG report: Tool reporting procedure (AHG13),” 13th JVET Meeting, Doc. JVET-M0013, Jan. 2019.
[5] C. Hsu and M. Winken, “BoG report on CE10 related contributions,” 13th JVET Meeting, Doc. JVET-M0873, Jan. 2019.
[6] X. Wang, Y. Chen, X. Xiu, and T. Ma, “CE4-related: An improved method for triangle merge list construction,” 14th JVET Meeting, Doc. JVET-N0340, March 2019.
[7] D. Park, Y. Yoon, J. Kim, J. Lee, and J. Kang, “CE10-related: Simplification of triangular partitions,” 13th JVET Meeting, Doc. JVET-M0352, Jan. 2019.
[8] F. Bossen, J. Boyce, X. Li, V. Seregin, and K. Suhring, “JVET common test conditions and software reference configurations for SDR video,” 12th JVET Meeting, JVET-L1010, Oct. 2018.