MODIFICATION OF THE MERGE CANDIDATE LIST FOR DEPENDENT VIEWS IN 3D-HEVC

E-G. Mora, J. Jung

Orange Labs, 38 Rue du G. Leclerc
92794 Issy Les Moulineaux, France
{elie.mora,joelb.jung}@orange.com

M. Cagnazzo, B. Pesquet-Popescu[*]

Telecom ParisTech, TSI Department
46 rue Barrault, 75634 Paris, France
{cagnazzo,pesquet}@telecom-paristech.fr

ABSTRACT

A test model for an HEVC-based 3D video coding standard (3D-HEVC) has recently been drafted. 3D-HEVC exploits inter-view redundancies by including disparity-compensated prediction (DCP) for efficient dependent view coding. It also uses the Merge coding mode to reduce the cost of motion / disparity parameters. However, the candidates in the Merge list are mostly temporal motion vectors. DCP does not often benefit from accurate predictors and is thus costly. Consequently, motion-compensated prediction (MCP) remains largely preferred.

In this paper, we propose to reduce the cost of DCP by modifying the Merge candidate list to always include a disparity vector candidate. Two methods are proposed: the new candidate is either added in the secondary or in the primary list of candidates. The latter method, which achieves average bitrate reductions of 0.6% for dependent views, and 0.2% for coded and synthesized views, was adopted in both the 3D-HEVC working draft and software.

Index Terms— 3D-HEVC, dependent view coding, disparity-compensated prediction, Merge candidate list

1. INTRODUCTION

New 3D multimedia services, such as 3D television [1] (3DTV) or free viewpoint television [2] (FTV) created a need for a 3D video standard that supports multiview video (MVV) and multiview video plus depth (MVD) formats. This need was answered with the release of an HEVC-based draft 3D coding standard (3D-HEVC) [3]. 3D-HEVC exploits spatial, temporal, inter-component and inter-view redundancies to efficiently encode the 3D video. Inter-view redundancies are in particular exploited by disparity-compensated prediction (DCP), currently present in the MVC standard [4]. DCP enables having, for the currently frame, reference frames from different views at the same time instant. The vector of a prediction unit (PU) pointing to a PU in a different view is called a disparity vector (DV), while a vector pointing to a reference frame in the same view but at a different time instant is called a motion vector (MV). Intensive work has been conducted on DCP and its combination with other tools to further increase coding efficiency [5, 6].

3D-HEVC uses the Merge coding mode [7] introduced in HEVC which establishes a list of candidate vector predictors to efficiently reduce the signaling cost of motion / disparity parameters (vectors + reference indices). This list rarely contains DV predictors, and although there is a multiview candidate in the list, it is preferred to be a MV than a DV. This MV / DV asymetry highly penalizes DCP, which remains largely less selected than motion-compensated prediction.

While numerous tools proposed in HEVC and in 3D-HEVC try to modify the Merge candidate list to achieve coding gains, before the 2$^{\text{nd}}$ JCT-3V meeting, no tools in 3D-HEVC were designed to populate the list with more DV candidates to reach a better equilibrium between DCP and MCP. In this paper, we propose to modify the Merge candidate list by inserting a new candidate which is always a DV. The DV candidate is inserted either in the secondary (method 1) or the primary (method 2) list of Merge candidates. Both methods were presented at the 2$^{\text{nd}}$ JCT-3V meeting and method 2 was adopted in both the 3D-HEVC working draft and the software [8].

The rest of this paper is organised as follows: Section 2 presents the Merge coding mode and its related state of the art. Section 3 describes our proposed method, and its coding results are given in Section 4. Section 5 concludes this paper, while underlining the possibilities for future work.

2. STATE OF THE ART

The Merge coding mode in 3D-HEVC allows a PU to inherit the motion / disparity parameters from a neighboring PU. Motion / disparity parameters from different neighboring PUs form the Merge candidate list. Only the index of the most coding efficient candidate is sent in the bitstream, along with an optional PU residual. Merge mode thus creates contiguous motion / disparity areas at a minimal cost in 4 different

[*]The authors acknowledge the support of the Persee project (ANR-09-BLAN-0170)

dimensions: horizontal, vertical, temporal, and inter-view.

3D-HEVC uses the Merge candidate list of HEVC [9], which consists, in order, of four spatial candidates and one temporal candidate. A pruning process is performed within the spatial candidates to remove redundant vectors [10]. 3D-HEVC adds a so-called multiview candidate, only for dependent views, in the first position of the list. If some of these 6 candidates are unavailable (the PU corresponding to the position falls outside the slice, or is Intra-coded, or the candidate is redundant), a secondary list of candidates is computed. These candidates are then appended to the list so that the total number of candidates is always 6. The candidates in that secondary list are, in order, combined candidates from mixed primary vectors of both reference lists, and zero-vector candidates, each having a different reference index.

The multiview candidate is computed in the following manner: first, a DV is derived for the current PU. In previous versions of 3D-HEVC, a depth map estimate was maintained for each view and the DV was derived from the highest depth value contained in the co-located PU in the estimate. Currently in 3D-HEVC, to reduce complexity, a simple neighbor search for a DV is performed. The DV allows finding a reference PU in the main view that corresponds to the current PU in the side view. The motion vector of that reference PU is then set as the multiview candidate. If the reference PU is intra-coded or falls outside the slice, the DV itself is set as the multiview candidate. Thus, there is always a temporal preference for this candidate, and consequently, the Merge list is in most cases composed only of MVs.

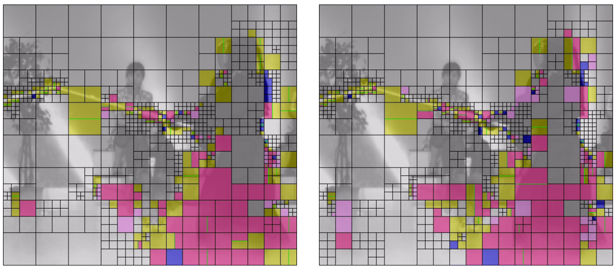
Several tools that modify the Merge candidate list construction in 3D-HEVC were proposed to either achieve coding gains, or reduce complexity / memory consumption: In [11], the primary candidate list is checked and the first DV candidate found is used to compute, by adding a positive and a negative offset, two more DV candidates which will then be added to the list. However this requires having a DV in the primary list to begin with, which is not a frequent case. Consequently, the coding gains are limited. The Merge pruning process can also be changed, like in [12], where a comparison between the inter-view candidate and the first two spatial candidates is added. This method achieved 0.3% bitrate reduction on dependent views with no runtime increase and was adopted in 3D-HEVC. For depth PU coding, the first Merge candidate was modified in [13] to refer to merging with the co-located texture PU, as the texture and depth motion information are highly correlated. A 1.1% bitrate reduction was reported for coded and synthesized views. Hence, this Motion Vector Inheritance tool was adopted in 3D-HEVC.

Tools that affect the Merge candidate list construction were also proposed in HEVC. In [14], the temporal candidate (TMVP) position is changed from the center of the co-located PU to the bottom-right position. Significant bitrate reductions of 0.9% were reported and thus the method was adopted in HEVC. In [15], two refined candidates were computed from the first Merge candidate and added to the secondary list of candidates, to replace the combined ones for uni-predicted PUs. Coding gains were not significant enough however to favor adoption.

These methods try to improve the candidate list construction but with no particular intention to balance the DCP selection against the MCP selection in the process. We propose, as described in the next section, a novel method to reach a better DCP / MCP equilibrium by inserting a DV candidate in the Merge list.

3. PROPOSED METHOD

Our work is based on the observation that DCP is not often selected by the HTM encoder. Also, Merge mode was observed to be often selected for coding PUs. This can be seen in Figure 1(a) which shows parts of a B-frame of the Kendo sequence coded with the reference HTM encoder. The PUs coded using MCP are shown in grey (Merge-SKIP) and green (Inter). PUs coded using DCP are shown in light pink (Merge-SKIP) and dark pink (Inter). Blue PUs are coded in Intra. We can clearly see that Merge mode is selected often, and that DCP coded PUs are not numerous. Table 1 gives the percentages of Merge coded PUs, DCP coded PUs, and DCP coded PUs in Merge mode, in dependent texture and depth views, averaged across four QPs, of seven MPEG sequences. These



(a) Reference (b) Proposed method

Fig. 1. CU coding modes in parts of a Kendo B-frame with a reference coding and with our proposed method

Sequence	Merge	DCP	DCP-Merge
Kendo	92	17	14
Newspaper	88	15	12
Balloons	93	13	11
Dancer	90	26	21
GT Fly	96	18	15
Poznan Hall2	95	9	7
Poznan Street	94	15	12
Average	**92**	**16**	**13**

Table 1. Percentage of Merge coded PUs, DCP coded PUs, and DCP coded PUs in Merge mode

results confirm the assertion that Merge mode is selected often, actually for 92% of PUs. This is due to the fact that Merge mode is very efficient at reducing the cost of motion / disparity parameters as only an index is encoded. Table 1 also shows that only 16% of PUs use DCP, and they are also most often coded in Merge mode (13%). While it is true that there are often more temporal correlations than inter-view, as shown in [16], the main issue behind the unfrequent DCP selection remains the lack of accurate DV predictors. Indeed, DCP can yield a better prediction for a given PU than MCP, in case there is little disparity between views or if there is fast motion in the video. However, not having a DV predictor in the Merge list increases the rate needed to code the PU with DCP since the only option left is to send a motion vector residual. MCP, while maybe not yielding a lower distortion value, requires a lower rate due to the fact that there are numerous MV predictors in the Merge list and signaling the motion parameters only costs an index. Consequently MCP is chosen more often since its Lagrangian cost is smaller, but if a DV predictor was added in the Merge list, as proposed in this work, the required rate for DCP coding would be decreased, hence increasing the selection of DCP and achieving coding gains.

When computing the multiview candidate in the Merge list, a DV pointing to a reference block in the base view is derived, as explained in Section 2. The multiview candidate is set as the MV of that reference block, and if that MV does not exist, it is set as the DV. We propose to insert that DV as a new interview candidate in the Merge list along side the multiview candidate if the latter turned out to be a MV.

Two insertion methods are proposed. In method 1, the candidate is inserted in the secondary list along with the combined and the zero vector candidates. If any of the first five candidates in the primary list is unavailable, the interview candidate is inserted after the final spatial candidate (before the temporal) to complete the list. If more primary candidates are unavailable, the combined and zero vector candidates are then appended to the list, as it is normally done. In method 2, the candidate is inserted in the primary list, in the 5th position, shifting the final spatial candidate to the 6th position. The temporal candidate is hence pushed out of the primary list and into the secondary list. It is the first candidate in the secondary list to be appended back in the primary list if some candidates are unavailable. Figure 2 illustrates these two methods. In both methods, before inserting the interview candidate, a redundancy check with all candidates preceding it in the list is performed for better coding efficiency. Note that the insertion positions in both methods have been empirically set as those positions gave out the most coding gains on average.

4. EXPERIMENTAL RESULTS

We have implemented our two proposed methods in HTM-4.1 [17]. We have strictly followed the common test conditions (CTCs) defined by JCT-3V [18]. A GOP of 8 was considered with an Intra period of 24. Four QP combinations for texture and depth (respectively) were considered: (25;34), (30;39), (35;42) and (40;45) to conform to CTCs. We have tested the two methods on seven sequences defined in the CTCs (1920×1088 and 1024×768). Experiments were done on 10 seconds of video length. Each sequence is composed of 3 texture and 3 depth views (one central base view and two side views). After encoding, 3 intermediate views were synthesized between the left and the center view, and another 3 between the center and the right views. PSNR on synthesized views were computed with respect to synthesized views rendered with uncompressed original texture and depth views. Coding gains are measured with the Bjontegaard delta (BD-Rate) metric [19].

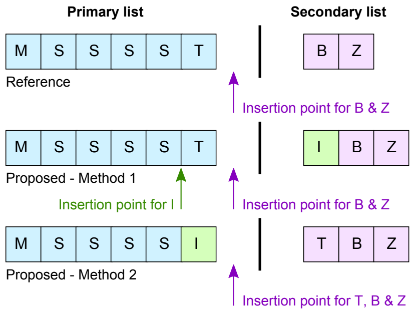


Fig. 2. Proposed insertion methods (M: multiview, S: spatial, T: temporal, B: combined, Z: zero, I: interview candidate)

Tables 2 and 3 give the coding gains (negative values are gains) and runtimes obtained with methods 1 and 2 respectively. These results are summarized in Table 4 which also gives the average results if the redundancy check preceding the insertion of the interview candidate in the list is removed. In these tables, the “Video” column shows the gains on the central (0) and on the two side views (1 and 2) and averages these results. The “Synt.” column gives results on the 6 synthesized views (the bitrate considered is the sum of the 3 texture and depth bitrates, and the PSNR is the average PSNR of all 6 synthesized views). The “Coded+Synt.” result is the same as in the previous column except that the PSNR considered is the average PSNR of the 6 synthesized views and the 3 coded texture views.

Tables 2 and 3 show bitrate reductions of 0.5% (resp. 0.6%) and 0.6% for side views, 0.2% for synthesized and 0.2% for coded and synthesized views. This is accompanied by a 3% (resp. 4%) encoder runtime reduction. No gains are achieved on the central view since our method is not applied there. Table 4 shows that coding efficiency is reduced if the redundancy check is removed, with no decrease in encoder and decoder runtimes compared to the original version.

The gains obtained result from an increase in DCP selection. Inserting a DV into the Merge candidate list reduces the rate needed for DCP coding and favors its selection, especially if there is small disparity between views (interview

Sequence	**Video**				**Synt.**	**Coded**	**Runtimes**	
	0	**1**	**2**	**Avg**		**+Synt**	**Enc**	**Dec**
Balloons	0.0	-0.6	-0.6	-0.3	-0.2	-0.2	96	102
Kendo	0.0	-0.6	-0.4	-0.2	-0.1	-0.2	100	101
Newspaper	0.0	-0.3	-0.3	-0.1	-0.1	-0.1	100	101
GT Fly	0.0	-1.2	-1.0	-0.3	-0.2	-0.3	97	100
Poznan Hall2	0.0	0.2	-0.5	-0.1	-0.1	-0.1	97	94
Poznan Street	0.0	-0.6	-0.6	-0.2	-0.2	-0.2	90	100
Dancer	0.0	-0.6	-0.6	-0.2	-0.2	-0.2	97	102
Average	**0.0**	**-0.5**	**-0.6**	**-0.2**	**-0.2**	**-0.2**	**97**	**100**

Table 2. Bitrate reduction per sequence, in %, with method 1

Sequence	**Video**				**Synt.**	**Coded**	**Runtimes**	
	0	**1**	**2**	**Avg**		**+Synt**	**Enc**	**Dec**
Balloons	0.0	-0.6	-0.6	-0.3	-0.2	-0.2	97	95
Kendo	0.0	-0.6	-0.4	-0.2	-0.1	-0.1	98	101
Newspaper	0.0	-0.3	-0.2	-0.1	-0.1	-0.1	100	90
GT Fly	0.0	-1.2	-1.2	-0.4	-0.2	-0.3	101	100
Poznan Hall2	0.0	-0.1	-0.3	-0.1	-0.1	-0.1	93	107
Poznan Street	0.0	-0.7	-0.6	-0.2	-0.2	-0.2	92	100
Dancer	0.0	-0.6	-0.6	-0.2	-0.2	-0.2	93	101
Average	**0.0**	**-0.6**	**-0.6**	**-0.2**	**-0.2**	**-0.2**	**96**	**99**

Table 3. Bitrate reduction per sequence, in %, with method 2

redundancies are much higher, and DVs can point to a better hypothesis) or if there is fast motion in the video (MVs are not able to correctly predict PUs). Figure 1(b) indeed shows an increase in DCP coded PUs compared to Figure 1(a). This is confirmed in the numerical results of Table 5, which shows, for both methods, an increase of 8% and 11% on average in the percentage of DCP-coded PUs and DCP-coded PUs using Merge mode.

The complexity resulting from the redundancy check used in the two methods is debatable. The purpose of this redundancy check is to avoid having a redundant DV candidate in the list which will either push potentially better primary candidates further down the list, while increasing their indices, and hence their signaling cost, in the process, or take the place of other, potentially better, secondary candidates which will not even be evaluated. The maximum number of checks equals 4 and 5 in method 1 and 2 respectively. These would be quite complex to perform for each PU. However, we show in Table 4 that removing the redundancy check decreases coding efficiency, as expected, while not reducing neither encoder or decoder runtime. Indeed, the worst case rarely occurs. Consequently, keeping the redundancy check is a better choice.

Method	**Video**				**Synt.**	**Coded**	**Runtimes**	
	0	**1**	**2**	**Avg**		**+Synt**	**Enc**	**Dec**
M1	0.0	-0.5	-0.6	-0.2	-0.2	-0.2	97	100
M1-NO RC	0.0	-0.4	-0.4	-0.1	-0.1	-0.1	98	101
M2	0.0	-0.6	-0.6	-0.2	-0.2	-0.2	96	99
M2-NO RC	0.0	-0.5	-0.4	-0.2	-0.1	-0.1	98	100

Table 4. Bitrate reductions when the redundancy check is removed (NO RC) in method 1 (M1) and 2 (M2)

Sequence	**DCP increase**		**DCP-Merge increase**	
	M1	**M2**	**M1**	**M2**
Kendo	6.2	6.4	9.1	8.9
Newspaper	5.0	4.9	7.8	8.0
Balloons	8.2	7.8	12.2	11.3
Dancer	7.7	7.3	10.8	10.6
GT Fly	17.6	17.0	21.0	20.3
Poznan Hall2	6.3	6.8	8.4	8.9
Poznan Street	6.9	8.2	9.3	10.8
Average	**8.3**	**8.3**	**11.2**	**11.3**

Table 5. Percentage increase of DCP-coded PUs and DCP-coded PUs using Merge mode in the two methods

The two methods also brought small encoder runtime reductions of 3 and 4%. This is because inserting a DV candidate in the Merge list means constructing one less secondary candidate, which is a complex process since it involves mixing different vectors to construct combined candidates or looping around all reference indices to construct zero-vector candidates. Additional experiments have shown that the number of constructed secondary candidates has decreased by 9% in the two methods.

5. CONCLUSION

In this paper, we have presented a novel method to improve the selection of DCP for dependent views in 3D-HEVC. A DV candidate has been inserted in the Merge candidate list in order to reduce the rate required for DCP, hence favoring its selection. Two insertion methods have been proposed, one where the DV is inserted in the secondary candidate list and another where the DV is inserted in the primary list. Bitrate reductions of 0.5% (resp. 0.6%) and 0.6% for the two side views, along with 0.2% for synthesized and for coded+synthesized views are reported. These were accompanied by a 3% (resp. 4%) encoder runtime reduction since secondary candidates are less required to be constructed. Both methods were presented at the 2nd JCT-3V meeting and method 2 was adopted in 3D-HEVC.

The gains obtained highly depend on the quality of the derived DV. The DV derivation process is the same as the one used for the multiview candidate. Improving this process can lead to coding gains due to the improvement of the multiview candidate which depends on it, but also due to the improvement of our newly added interview candidate. Hence, it is an interesting topic for future work.

6. REFERENCES

[1] C. Fehn, E. Cooke, O. Schreer, and P. Kauff, "3D analysis and image-based rendering for immersive TV applications," *Signal Processing: Image Communication*, vol. 17, no. 9, pp. 705 – 715, 2002.

[2] J. Carranza, C. Theobalt, M. Magnor, and H-P. Seidel, "Free-viewpoint video of human actors," *ACM Trans. Graph.*, vol. 22, no. 3, pp. 569–577, July 2003.

[3] G. Tech, K. Wegner, Y. Chen, and S. Yea, "Draft of 3D-HEVC test model description," ITU-T SG 16 WP 3 & ISO/IEC JTC 1/SC 29/WG 11 JCT3V-B1005, October 2012.

[4] Y. Chen, Y.-K. Wang, K. Ugur, M. Hannuksela, J. Lainema, and M. Gabbouj, "The emerging MVC standard for 3D video services," *EURASIP Journal on Advances in Signal Processing*, 2009.

[5] B. Girod, C-L Chang, P. Ramanathan, and X. Zhu, "Light field compression using disparity-compensated lifting," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2003, vol. 4, pp. 760–763.

[6] E. Martinian, A. Behrens, J. Xin, A. Vetro, and H. Sun, "Extensions of H.264/AVC for multiview video compression," in *IEEE International Conference on Image Processing*, October 2006, pp. 2981 –2984.

[7] P. Helle, S. Oudin, B. Bross, D. Marpe, M. Bici, K. Ugur, J. Jung, G. Clare, and T. Wiegand, "Block merging for quadtree-based partitioning in HEVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1720 –1731, December 2012.

[8] E. Mora, J. Jung, B. Pesquet-Popescu, and M. Cagnazzo, "3D-CE5.h related: Modification of the merge candidate list for dependant views in 3DV-HTM," ITU-T SG 16 WP 3 & ISO/IEC JTC 1/SC 29/WG 11 JCT3V-B0069, October 2012.

[9] B. Bross, W-J. Han, J-R. Ohm, G. Sullivan, and T. Wiegand, "High Efficiency Video Coding (HEVC) text specification draft 9," ITU-T SG16 WP3 & ISO/IEC JTC 1/SC 29/WG 11 JCTVC-K1003, October 2012.

[10] O. Bici, J. Lainema, and K. Ugur, "Non-CE13: Simplification of merge mode," ITU-T SG16 WP3 & ISO/IEC JTC 1/SC 29/WG 11 JCTVC-G593, November 2011.

[11] T. Guionnet, L. Guillo, and C. Guillemot, "CE5.h: Merge candidate list for disparity compensated prediction," ITU-T SG 16 WP 3 & ISO/IEC JTC 1/SC 29/WG 11 JCT3V-B0080, October 2012.

[12] J-L. Lin, Y-W. Chen, Y-W. Huang, and S. Lei, "3D-CE5.h: Pruning process for the inter-view candidate," ITU-T SG 16 WP 3 & ISO/IEC JTC 1/SC 29/WG 11 JCT3V-B0086, October 2012.

[13] M. Winken, H. Schwarz, and T. Wiegand, "Motion vector inheritance for high efficiency 3D video plus depth coding," in *Picture Coding Symposium (PCS)*, May 2012, pp. 53 –56.

[14] J-L. Lin, Y-W. Chen, Y-P. Tsai, Y-W. Huang, and S. Lei, "Motion vector coding techniques for HEVC," in *IEEE 13th International Workshop on Multimedia Signal Processing (MMSP)*, October 2011, pp. 1 –6.

[15] T. Yamamoto and T. Ikai, "Merge candidate refinement for uni-predictive block," ITU-T SG16 WP3 & ISO/IEC JTC 1/SC 29/WG 11 JCTVC-I0293, May 2012.

[16] Y. Zhang, G. Yi Jiang, M. Yu, and Y. Sung Ho, "Adaptive multiview video coding scheme based on spatiotemporal correlation analyses," *ETRI Journal*, vol. 31, no. 2, April 2009.

[17] G. Tech, "HTM-4.1 software," Available: https://hevc.hhi.fraunhofer.de/svn.

[18] D. Rusanovsky, K. Muller, and A. Vetro, "Common Test Conditions of 3DV Core Experiments," ITU-T SG 16 WP 3 & ISO/IEC JTC 1/SC 29/WG 11 JCT3V-A1100, July 2012.

[19] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," VCEG-M33, April 2001.