Depth Video Coding Based on Motion Information Sharing Prediction for Mixed Resolution 3D Video

Kwan-Jung Oh[*] and Ho-Cheon Wey[†]
[*] Broadcasting & Telecommunications Media Research Laboratory, ETRI, Daejeon, Rep. of Korea
E-mail: kjoh@etri.re.kr
[†] Multimedia R&D Team, DMC R&D Center, Samsung Electronics, Suwon, Rep. of Korea
E-mail: hc.wey@samsung.com

Abstract— **Three-dimensional (3D) video has evolved and spread rapidly in recent years. In the 3D video, since a depth component and corresponding texture component have similar object movement, there is a redundancy in their motion information. To remove this motion redundancy, motion information sharing from texture to depth is useful in 3D video coding. In this paper, we propose a method for motion information sharing prediction (MISP) for use in mixed resolution 3D video coding. The MISP is applied for any types of macroblock even it is coded as intra mode. Experimental results show that the proposed MISP reduces the bit rate for depth by up to 7.3%.**

I. Introduction

The three-dimensional (3D) video related technologies have been developed rapidly and 3D age is much closer to us. What started as simple, stereoscopic video has extended to multiview video (MVV) to allow free-viewpoint rendering. However, the MVV system does not fully represent infinite views because of the limited number of views. Thus, multiview video plus depth (MVD) system has been developed. The MVD system can handle the infinite views by synthesizing intermediate views with the depth image based rendering (DIBR) technique [1].

Given the large amount of data associated with MVV and MVD, the need for efficient 3D video coding techniques has become acute. MVV is typically coded with a multiview video coding (MVC) extension of H.264/MPEG-4 advanced video coding (AVC) standard jointly developed by the ITU-T video coding experts group (VCEG) and the ISO/IEC moving picture experts group (MPEG). To support the MVD format, MPEG issued a call for proposals (CfP) for 3D video coding in March 2011. At the 98th MPEG meeting held in Gevena, various H.264/AVC compatible and HEVC (high efficiency video coding) compatible proposals were evaluated and now both AVC-based and HEVC-based 3D video coding have been studied.

The 3DV-AVC codec supports the mixed resolution 3D video consisting of full-size texture video and reduced-size depth video. It also supports flexible coding order for the texture and depth components. The base texture view is coded independently from the depth component and other texture views so that it remains compatible with H.264/AVC. Various inter-view, inter-component, and depth coding tools have been developed as a part of 3DV-AVC. For inter-view prediction, view synthesis prediction (VSP) which uses the synthesized virtual view as a reference frame was proposed [2]. For the inter-component coding, depth-based motion vector prediction (DMVP) was proposed [3]. A synthesized view distortion was proposed for depth coding. It is used for rate-distortion optimization (RDO) instead of depth distortion to improve rendering quality [4]. The researches on efficient depth boundary coding [5] and boundary reconstruction filter [6] also have been studied.

To efficiently remove the motion redundancy between texture and depth components, a depth coding method based on motion information sharing prediction (MISP) is proposed in this paper. Although inside view motion prediction (IVMP) has already been adopted in 3DV-AVC as a motion information sharing tool, it is hampered by problems involving mixed resolution 3D video coding. In general, the motion information of a texture macroblock has been shared by the depth macroblock, since the texture video has more accurate motion vectors. A depth macroblock coded by a motion information sharing prediction typically copies all types of motion information, including macroblock type, sub-macroblock type, motion vectors, and reference indices, from its corresponding texture macroblock.

Various methods for sharing motion information between texture and depth have been studied in the field [7]-[8], but these assume that the 3D video will use the same resolution for both texture and depth. They must therefore be modified to support mixed resolution 3D video, allowing one depth macroblock to correspond to multiple texture macroblocks. This modification may present problems when the depth macroblock has a corresponding texture macroblock coded in intra and P8x8 mode. An intra-coded macroblock does not have motion information, and H.264/AVC does not support motion vectors for a sub-block of less than size 4x4, as in the case of P8x8 mode. Thus, existing IVMP is signaled and enabled for inter-coded macroblocks only, except for P8x8 mode, thus presenting a parsing dependency problem that will degrade coding efficiency and increase the decoding complexity. Our proposed motion information sharing prediction (MISP) method solves the above problems by allowing motion information sharing even when the corresponding texture macroblock is coded in intra and P8x8 mode.

II. Overview of MVD Format and 3D-AVC

The MVD system illustrated in Fig. 1 is an extension of the MVV system that associates texture data with its corresponding depth data.

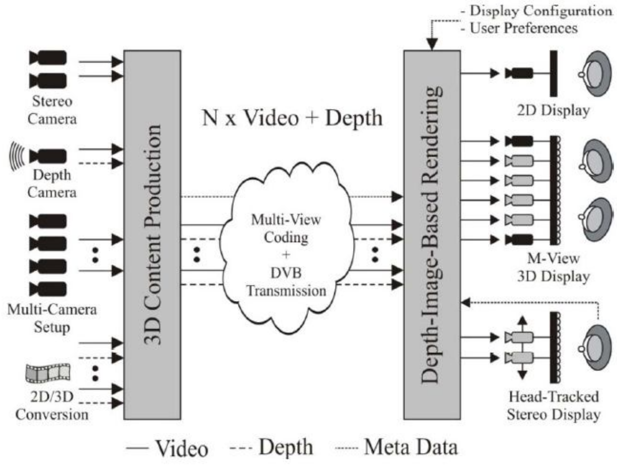

Fig. 1 Multiview video plus depth (MVD) system.

The MVD data of a single view consists of a 4:2:0 texture image and a 4:0:0 depth image, as shown in Fig. 2. In general, the number of views for depth is either the same as or less than the number of views for texture. To reduce the size of MVD data further, a mixed resolution MVD format, shown in Fig. 3, has been proposed. The 3DV-AVC codec supports both mixed resolution and matching resolution 3D video. In general, the resolution of the depth is reduced, since it less sensitive to down-sampling.

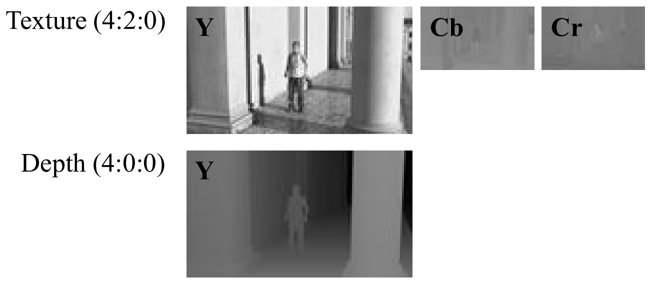

Fig. 2 Image format of MVD data.

According to 3DV-AVC, the view containing the first coded I frame is the base view, and is compatible with H.264/AVC. Other views are known as dependent views. For the base view, the texture is coded first and the depth second, whereas for dependent views, all depths are all coded first, followed by all textures. For example, in a simplified 3-view case, coding order will be T1-D1-D0-D2-T0-T2, as shown in Fig. 3. Thus, texture-based depth coding tools, such as motion information sharing prediction (MISP), are applied only to the base view, while depth-based coding tools, such as view synthesis prediction (VSP) and depth based motion vector prediction (DMVP), are applied to dependent views [9]-[10].

Fig. 3 Mixed resolution MVD format.

III. Motion Information Sharing Prediction (MISP)

In video coding, high compression efficiency is achieved by using motion estimation and motion compensation processes. The temporal redundancy between the successive frames is removed by estimating the motion between the current frame and the reference frame, and then generating a motion field to minimize temporal prediction error. MVD-based 3D video data usually contains twice as many motion vectors for texture components as for depth components. Even so, we can assume that texture and depth components will have high motion correlation within a scene, as illustrated in Fig. 4. We can also assume that, in general, motion information from texture sharing by depth coding is more reliable than motion information from depth sharing by texture coding.

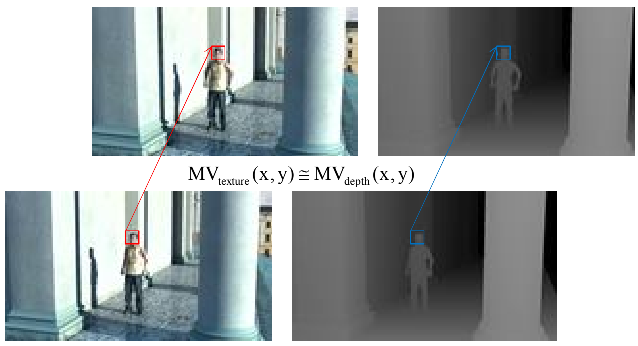

Fig. 4 Motion correlation between texture and depth.

In the mixed resolution 3D video coding, one depth macroblock has multiple texture correspondences since the depth resolution is lower than the texture resolution. Fig. 5 shows a macroblock correspondence of texture and depth for mixed resolution 3D video.

The efficiency of the motion information sharing method depends on its availability—i.e. how many depth macroblocks can be coded within the motion information sharing method. The proposed MISP method allows for as many depth macroblocks as possible. Furthermore, whereas the IVMP method only allows depth macroblocks with all inter-coded texture correspondences (excepting those in P8x8 mode), MISP allows macroblocks of any depth, including those with

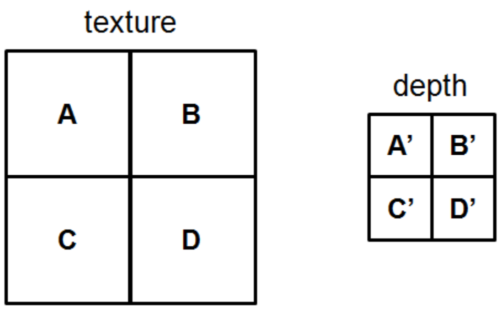

Fig. 5 Macroblock correspondence of texture and depth for mixed resolution 3D video.

all intra-coded texture correspondences. In MISP, intra-coded texture macroblocks are replaced by an inter-coded neighboring macroblock. The macroblock that has the most complex mb_type is used to replace the intra-coded macroblock when two or more neighboring inter-coded macroblocks are available. Fig. 6 shows the sample texture macroblocks that are available in MISP, but not in IVMP.

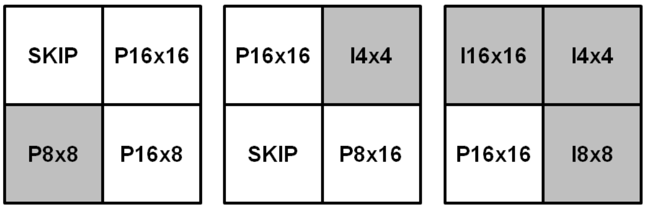

Fig. 6 Texture macroblocks that are available in MISP, but not in IVMP.

Motion information sharing is performed in each 8x8 sub-block. In Fig. 5, the texture macroblock B shares motion information with the 8x8 size depth block B'. MISP predicts the reference indices and motion vectors only. That is, it exploits an 8x8 size block for motion prediction regardless of the mb_type of the texture macroblock, whereas IVMP uses a block shape of half the size of the texture mb_type. This is because the 8x8 block size is sufficient to represent the object-dependent motion of depth and some exquisite motions of the texture can be erroneous for depth. Thus, MISP determines the most reliable motion information from several candidates, the number of which depends on the mb_type, and may reach as high as 16 when all 4x4 blocks have independent motion. We assume that motion information is more reliable as the magnitude of the motion vector increases. To minimize complexity, this magnitude is calculated using a simple formula (1).

$$|MV(x, y)| = |x| + |y| \tag{1}$$

MISP uses the reference indices and motion vectors of a maximum magnitude block. The scale of the motion vector is adjusted according to the resolution ratio between texture and depth. Fig. 7 shows an example of the motion predictions of IVMP and MISP for inter-coded texture MB in P8x16 mode.

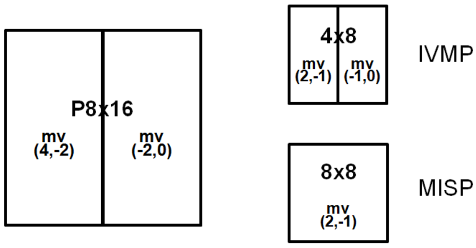

Fig. 7 Motion prediction comparison between IVMP and MISP.

For an intra-coded macroblock, MISP selects the most reliable block (i.e. the one with highest magnitude motion vector) among the nine neighboring blocks, as shown in Fig. 8. MISP then sets the reference index of the selected block as 0 of list0 and the motion vector of the block as (0, 0).

Fig. 8 Candidate neighboring blocks for intra coded macroblock.

Both IVMP and MISP are signaled by a macroblock level flag. This is located between mb_skip_flag and other modes and is coded with CABAC. Note that IVMP has a parsing dependency problem when decoding the bit-stream for the depth component. It sends an IVMP-specific flag when IVMP is available, but will not send any flag when the corresponding texture macroblocks are coded in intra or P8x8 modes. Thus, IVMP availability must be checked every time an IVMP flag is parsed, creating a dependency problem between texture and depth components and increasing decoder complexity. MISP does not share this parsing dependency problem since MISP is available regardless of texture macroblock type. Following the MISP flag, residual data is encoded as needed. The MISP macroblock has mb_type of P8x8 and its deblocking filtering process is the same as that for the P8x8 mode.

IV. Experimental Results

The proposed MISP was implemented in 3DV-ATM version 0.4 [11] and tested using a simulation under conventional setup [12]. All seven test sequences were run on a 3-view configuration and all adopted tools were enabled. The coding results are summarized and compared with IVMP in Table 1, using the BDBR (Bjøntegaard delta bit rate) metric [13] in terms of the total bit rate and the PSNR of decoded texture views, as well as the total rate and PSNR of rendered virtual views. The virtual views were synthesized using VSRS 1-D fast software [14].

The results in Table 1 indicate that MISP nearly doubled the performance of IVMP. The decoded and synthesized gains were stable for all sequences and did not depend on depth types and image resolutions. IVMP and MISP were applied to the base view only since the current 3DV-ATM coding order is T1-D1-D2-D3-T2-T3. We can expect better coding performance for a T1-D1-T2-D2-T3-D3 coding order. Table 2 presents the coding results for texture views and depth views. MISP achieves 3.60% BD gains in terms of depth bit-rate and depth PSNR and reduces the bit rate for depth by up to 7.3% for GT_Fly sequence. In addition, MISP also achieves 0.18% BD gains in terms of texture bit-rate and texture PSNR. It means that MISP give a positive effect to both depth and texture components.

V. Conclusions

The proposed motion information sharing prediction (MISP) is designed to reduce motion information redundancy between texture and depth for the mixed resolution 3D video. Because of its design, MISP is always available, regardless of the macroblock type of the corresponding texture, while conventional IVMP is disabled for P8x8 and intra modes. Only the block with the highest magnitude motion vector is selected from neighboring sub-blocks and used for motion information. Furthermore, MISP can be signaled with a macroblock level flag without the parsing dependencies found in IVMP. Finally, experimental results show that the proposed MISP achieves 0.60% BD gains in terms of total bit-rate and synthesized PSNR.

References

[1] C. Fehn, “Depth-image-based rendering (DIBR), compression and transmission for a new approach on 3D-TV,” in Proceedings of SPIE Stereoscopic Displays and Virtual Reality Systems, vol. 5, pp. 93–104, San Jose, USA, January 2004.

[2] S. Yea and A. Vetro, “View synthesis prediction for multiview video coding,” Signal Processing: Image Communication, vol. 24, no. 1, p. 89-100, Jan. 2009.

[3] ISO/IEC JTC1 SC29 WG11 MPEG, “Description of Nokias response to MPEG 3DV Call for Proposals on 3DV VIDEO Coding Technologies,” M22552, Nov. 2011.

[4] B. T. Oh, J. Lee, and D.-S. Park, “Depth map coding based on synthesized view distortion function,” IEEE Journal of Selected Topics in Signal Processing, vol. 5, no. 7, pp. 1344–1352, Nov. 2011.

[5] B. T. Oh, J. Lee, and D.-S. Park, “Plane segmentation based intra prediction for depth map coding,” in Picture Coding Symposium, pp. 41–44, 2012.

[6] K.-J. Oh, A. Vetro, and Y.-S. Ho, “Depth coding using a boundary reconstruction filter for 3D video system,” IEEE Transactions on Circuits System and Video Technology, vol. 21, no. 3, pp. 350–359, Mar. 2011.

[7] S. Grewatsch and E. Muller, “Sharing of motion vectors in 3D video coding,” in Proceedings of IEEE International Conference on Image Processing (ICIP ’04), vol. 5, pp. 3271–3274, Singapore, October 2004.

[8] H. Oh and Y.S. Ho, “H.264-based depth map sequence coding using motion information of corresponding texture video,”in Proceedings of the 1st Pacific Rim Symposium on Advances in Image and Video Technology (PSIVT ’06), pp. 898–907, Hsinchu, Taiwan, December 2006.

[9] ISO/IEC JTC1/SC29/WG11, “Working Draft 2 of AVC compatible video with depth information,” N12732, May 2012.

[10] ISO/IEC JTC1/SC29/WG11, “Test Model for AVC based 3D video coding,” N12558, March 2012.

[11] MPEG 3DV AVC-based 3DV reference software version 0.4: http://mpeg3dv.research.nokia.com/svn/mpeg3dv/trunk/

[12] ISO/IEC JTC1/SC29/WG11, “Common Test Conditions for 3DV experimentation,” N12745, May 2012.

[13] ITU-T SG16 Q.6, “An excel add-in for computing Bjøntegaard metric and its evolution,” VCEG-AE07, Jan. 2007.

[14] MPEG 3DV view synthesis reference software: VSRS 1-D Fast software.

Table I
Total Coding Results for IVMP and MISP

Sequences	total rate vs. decoded PSNR		total rate vs. synthesized PSNR	
	IVMP	MISP	IVMP	MISP
PoznallHall2	-0.06%	-0.34%	-0.25%	-0.42%
PoznanStreet	-0.38%	-0.52%	-0.44%	-0.58%
UndoDancer	-0.27%	-0.42%	-0.35%	-0.62%
GT_Fly	-0.34%	-0.45%	-0.68%	-0.78%
Kendo	-0.23%	-0.37%	-0.35%	-0.63%
Balloons	-0.23%	-0.41%	-0.44%	-0.60%
Newspaper	-0.32%	-0.38%	-0.49%	-0.56%
Average	-0.26%	-0.41%	-0.43%	-0.60%

Table II
Texture and Depth Coding Results for IVMP and MISP

Sequences	depth rate vs. depth PSNR		texture rate vs. texture PSNR	
	IVMP	MISP	IVMP	MISP
PoznallHall2	-2.19%	-1.91%	0.06%	-0.21%
PoznanStreet	-2.66%	-3.49%	-0.23%	-0.32%
UndoDancer	-3.20%	-4.61%	-0.14%	-0.21%
GT_Fly	-5.54%	-7.29%	-0.24%	-0.25%
Kendo	-1.32%	-1.67%	-0.03%	-0.10%
Balloons	-3.02%	-3.69%	0.00%	-0.11%
Newspaper	-2.22%	-2.56%	-0.04%	-0.05%
Average	-2.88%	-3.60%	-0.09%	-0.18%