Depth Map Coding Method for Scalable Video Plus Depth

Panpan Li[1], Ran Ma[1,2(✉)], Yu Hou[1], and Ping An[1,2]

[1] School of Communication and Information Engineering, Shanghai University, Shanghai 200072, China
maran@shu.edu.cn

[2] Key Laboratory of Advanced Display and System Application of the Ministry of Education, Shanghai 200072, China

Abstract. Depth map coding along with its associated texture video plays an important role in the display of 3D scene by Depth Image Based Rendering technique. In this paper, the inter-layer motion prediction in Scalable Video Coding is applied in order to utilize the similarity of the motion data between texture video and its associated depth map. Additionally, in order to preserve the edge in depth map, the edge detection algorithm is proposed in depth map coding, which combines the conventional Sobel edge detection and the block-based coding scheme of SVC. And the dynamic quantization algorithm is proposed to preserve the information of boundary regions of the depth map. Simulation results show that the proposed method achieves the BDPSNR gains from 0.463 dB to 0.941 dB for the depth map, and the Bjøntegaard bitrates savings range from 9.72 % to 19.36 % for the video plus depth.

Keywords: 3D video · Video plus depth · Scalable Video Coding · Boundary region · Inter-layer prediction · Edge detection

1 Introduction

With the rapid development of the Internet and video display devices, the video multimedia services have attracted great interest of people. Compared with the traditional 2D video, 3D video can provide more realistic scene. Hence, the research of 3D video is regarded as a hot area and lots of progresses have been made. As one of the 3D video formats [1, 2], texture video plus depth map (V + D) is able to provide 3D scene by Depth Image Based Rendering (DIBR) technique [3], which can synthesis a virtual view based on a texture video and its associated depth map. Accordingly, it is quite necessary to transmit the depth map in addition to the texture video.

As an extension of the H.264/AVC standard [4], Scalable Video Coding (SVC) can provide some significant functionalities by dividing the bit stream into one base layer (BL) bit stream and several enhancement layers (ELs) bit streams [5]. In the process of transmission, the decoder can get an optimal bit stream through selectively discarding enhancement layer bit streams. In order to provide scalable 3D scene in the display side, the texture video and its associated depth map should be transmitted to the display device simultaneously. One simplest way is to encode them separately in the SVC

Y.-J. Zhang (Ed.): ICIG 2015, Part II, LNCS 9218, pp. 103–115, 2015.
DOI: 10.1007/978-3-319-21963-9_9

encoder and then synthesis the 3D video in the display side. However, this method doesn't exploit the correlation between texture video and its associated depth map. More redundant information could be removed and the compression efficiency could be further improved when they are jointly encoded.

In the process of 3D video coding, it can effectively improve the coding efficiency of the multi-view video transmission by adding the depth map. So, there were many studies about depth map coding. In [6], Tao et al. proposed a novel method of compressing the texture video and depth map jointly by using the correlation of the motion field and brought significant coding gain for the associated depth map. In [7], Lei et al. proposed a new depth map coding scheme based on the structure and motion similarities between the texture video and its associated depth map, a new type of block named OD-Block and a DTCC based prediction method are presented for the depth map coding, which can improve the coding efficiency and the rendering quality. The similarity of motion vectors between texture video and depth map were also used for the depth map coding in [8, 9].

Besides, some new edge-preserving algorithms were proposed for the reason that boundary regions play an important role in view rendering [10–13]. In [10], Zamarin et al. proposed a new edge-preserving intra mode for depth macroblocks (MBs) to improve the compression efficiency and the rendering quality. Kao and Tu proposed a novel compression method which includes an edge detection module, a homogenizing module and a compression encoding module for depth map in [11].

However, those methods mentioned above didn't combine the importance of the depth map edge and the correlation between texture video and its associated depth map when encoding the texture video and depth map in a scalable way. In this paper, a depth map coding method in the SVC is studied. A new depth map edge detection algorithm and a dynamic quantization algorithm based on the edge regions are proposed.

The rest of this paper is organized as follows. First, in Sect. 2, the proposed single-view video plus depth map coding method is described in detail. Then, the simulation results are shown in Sect. 3. Finally, Sect. 4 concludes the paper with a summary.

2 The Proposed Method

The depth map corresponding to the texture video, as shown in Fig. 1, can be regarded as a special color image whose pixel value represents the distance between the camera and the object. Besides, the depth map consists of two parts, namely boundary region (BR) and homogeneous region(HR). Specifically, the depth map is used for view rendering, rather than being displayed in the display devices.

Compared with the SVC scheme of the 2D texture video, there's no doubt that it's very worthy of studying the depth map coding method in the SVC scheme of video plus depth. So in this section, the SVC scheme of video plus depth, which is compatible with quality scalable and stereoscopic display, is firstly presented and the inter-layer prediction is also discussed. Then, the quantization algorithm based on edge detection for depth map coding is analyzed and studied to improve the quality of the depth map.

(a) (b)

Fig. 1. The video plus depth format of 3D video: (a) the texture video; (b) the depth map.

2.1 SVC Scheme of Video Plus Depth

Tao et al. in [6] encoded the single-view video plus depth map into two layers, one is the BL whose input is the texture video, and the other is the EL whose input is the depth map. Based on Tao's scheme, the improved encoding scheme proposed in this paper is illustrated in Fig. 2. Here, the three-layer coding structure is used.

In the BL, the hybrid temporal and spatial prediction mode is used to encode the texture video to get the basic video quality in Fig. 2. EL1, the newly added layer in our structure, is a texture quality enhancement layer. The data in EL1 is the refinement of texture information. When the decoder gets BL or/and EL1 bit stream, 2D video with different qualities can be displayed. In EL2, the depth map is input. When the decoder gets the whole bit stream that includes the BL, EL1 and EL2, 3D scenes are available by using the DIBR technique. Therefore, compared with Tao's scheme, our encoding scheme can realize not only the quality scalable but also stereoscopic display.

In the SVC scheme, inter-layer prediction is used, which is a prediction tool added to spatial SVC [5]. Specifically, inter-layer prediction can be divided into three types: inter-layer motion prediction, inter-layer residual prediction and inter-layer intra prediction. When encoding the enhancement layer, if the co-located MB in the reference

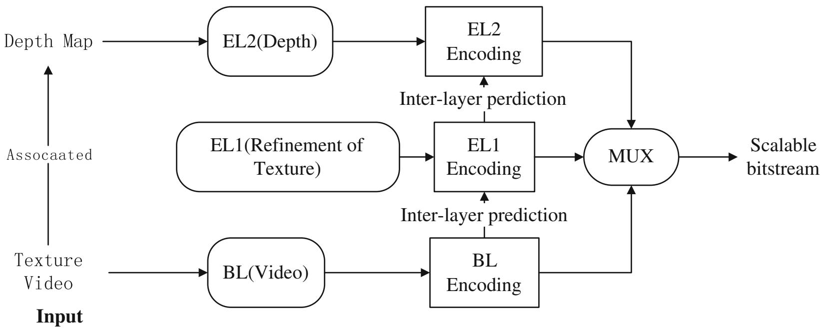


Fig. 2. The encoding scheme.

layer is inter-coded, then the current MB is also inter-coded, and its motion vectors and reference indexes can be derived from the co-located MB in the reference layer by inter-layer motion prediction. Besides, quality SVC can be considered as a special case of spatial SVC with the same video sizes for BL and ELs, so the inter-layer prediction can also be applied for it.

When encoding the EL1, the above three inter-layer prediction tools are all applied to improve the coding efficiency. In fact, it is the 2D quality SVC with BL and EL1, and the coding scheme is the same as the H.264/SVC, so it isn't discussed in this paper. When encoding the EL2, only the inter-layer motion prediction is applied to improve the coding efficiency of the depth map. There are several reasons for this. Firstly, it is easy to see that the structure between the texture video and the depth map is very similar from Fig. 1. Secondly, because the texture video and its associated depth map are shot in the same place and at the same time, they may have similar motion vectors. Actually, lots of researchers have already verify the motion similarity between them and the coding efficiency is improved a lot by using the similarity, such as the papers mentioned above [6–9]. Thirdly, the inter-layer residual prediction and inter-layer intra prediction are not applied because the residual information and pixel value of the depth MB is quite different from the texture MB.

2.2 Depth Map Edge Detection

Due to the effect of BR of the depth map on the quality of the synthesized view, the boundary data should be preserved as exactly as possible in the process of encoding. Therefore, for the depth map encoding, we should first extract the boundary area from the depth map. The edge preserving methods mentioned above are all based on the pixel. However, in the H.264/SVC standard, the basic encoding unit is MB and in the process of mode decision, the sizes of the blocks that compose of the MB partition modes can be divided into 16×16, 8×8, and 4×4. Therefore, designing an edge detection algorithm based on blocks of different sizes is more suitable for block based coding scheme than the pixel based edge detection algorithm. So in this paper, a block-based Sobel (BBS) edge detection algorithm is proposed to extract the boundary region. The basic idea of the proposed BBS algorithm is as follows:

The current MB has multiple prediction modes with the block size of $N \times N$ $(N = 16, 8, 4)$ in the process of encoding. In $N \times N$ block, each pixel's horizontal and vertical gradient can be calculated according to the conventional Sobel edge detection algorithm and then the joint gradient $G(x, y)$ can be obtained by the following formula (1):

$$G(x, y) = \sqrt{G_x(x, y)^2 + G_y(x, y)^2} \tag{1}$$

where, $G_x(x, y)$ and $G_y(x, y)$ represent horizontal and vertical gradient, (x, y) is the coordinate in $N \times N$ block.

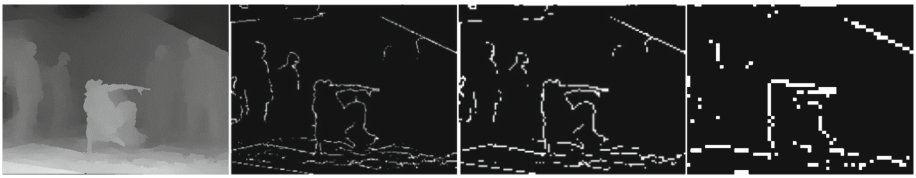
For $N \times N$ block, the average gradient value of $(x, y)(x, y \in [1, N-2])$ is set as the block-gradient of current $N \times N$ block. So, the block-gradient of $N \times N$ block can be expressed by formula (2):

$$G_N = \frac{\sum_{x=1}^{N-2}\sum_{y=1}^{N-2} G(x, y)}{(N-2)^2} \tag{2}$$

After obtaining the block-gradient of $N \times N$ block, it is compared with a threshold (*Thr*), which is set to 15 based on the empirical value. If $G_N > Thr$, then the current $N \times N$ block is in the BR, otherwise it's in the HR. *Breakdancers* sequence and *Ballet* sequence are experimented with the proposed BBS edge detection algorithm, and the results are shown in Figs. 3 and 4.

(a) (b) (c) (d)

Fig. 3. Results of *Ballet* sequence: (a) the original depth map; (b) BR with 4 × 4 block size; (c) BR with 8 × 8 block size; (d) BR with 16 × 16 block size.



(a) (b) (c) (d)

Fig. 4. Results of *Breakdancers* sequence: (a) the original depth map; (b) BR with 4 × 4 block size; (c) BR with 8 × 8 block size; (d) BR with 16 × 16 block size.

As we can see from Figs. 3 and 4, the proposed BBS edge detection algorithm can extract the BR of the depth map in a relatively accurate manner, and the smaller of the block size is, the more accurate the BR will be.

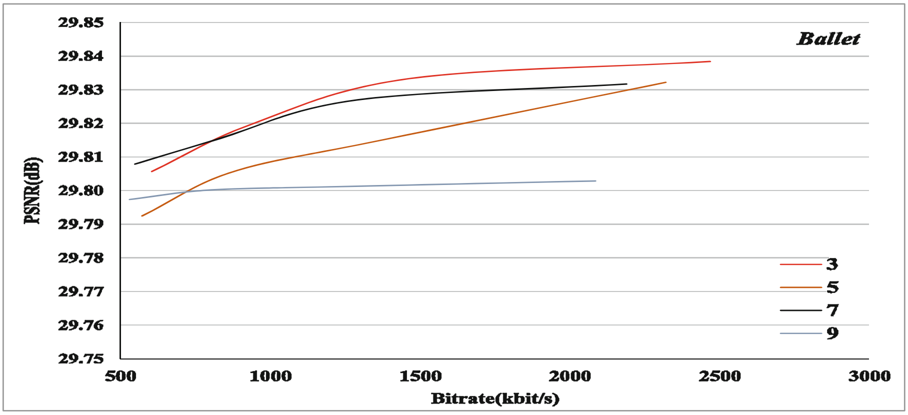
2.3 Dynamic Quantization

In the process of quantization, if the QP of the texture video has been decided, it's quite important to choose a suitable QP for the depth map to get a good quality of the synthetic view. When the bit rate of the depth map accounts for 10 %–20 % of the bit rate of the texture video, a good quality of the depth map can be gotten [14]. So some experiments are carried out to choose a better QP for the depth map based on the distribution of quantitative parameters for texture and depth. In our experiments, four groups of QP are selected for texture videos, (28,24), (32,28), (36,32) and (40,36) respectively. And in each group, the first parameter is for the BL and the second parameter is for the quality EL1. Three representative sequences are tested in our experiments, including *Ballet, Breakdancers* and *Newspaper*. The first two sequences have violent motion and the last one has slow motion. For the QP of the depth map, it is obtained by adding the QP Difference to the QP for the texture video of BL. In our experiments, we take the quality of the synthetic view as an evaluation criterion for the reason that the purpose of getting a high quality depth map is to get a high quality synthetic view. The experimental results can be seen in Table 1.

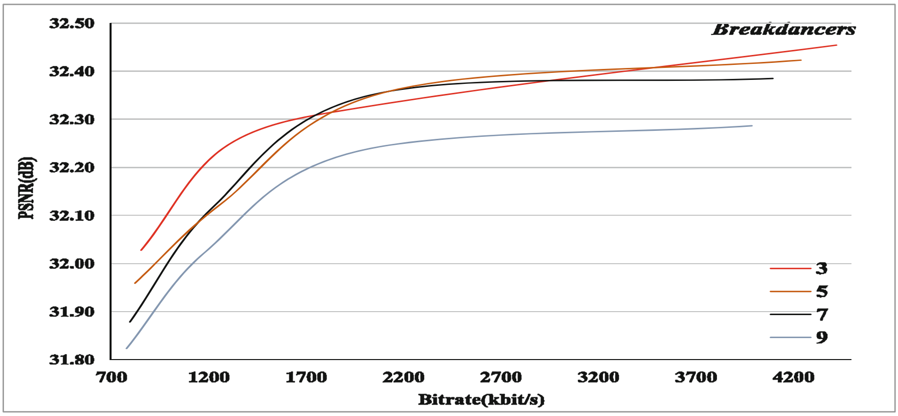
Table 1. Experimental results for selecting an optimal QP for the depth map encoding.

QP difference	QP	Sequences					
		Ballet		*Breakdancers*		*Newspaper*	
		Bitrates (kbps)	*SynView PSNR* (dB)	*Bitrates* (kbps)	*SynView PSNR* (dB)	*Bitrates* (kbps)	*SynView PSNR* (dB)
3	28-24-31	2469.010	29.8384	4422.182	32.4539	3615.360	28.1486
	32-28-35	1462.819	29.8333	2233.258	32.3399	2260.032	28.1421
	36-32-39	938.856	29.8198	1334.515	32.2544	1486.594	28.0958
	40-36-43	604.757	29.8057	855.922	32.0188	987.014	28.0607
5	28-24-33	2320.637	29.8322	4239.907	32.4227	3526.046	28.1025
	32-28-37	1359.854	29.8148	2124.653	32.3578	2202.134	28.0864
	36-32-41	874.402	29.8056	1274.107	32.1264	1453.238	28.0778
	40-36-45	572.472	29.7925	824.108	31.9590	968.962	28.0730
7	28-24-35	2189.789	29.8317	4095.979	32.3846	3451.411	28.1734
	32-28-39	1283.890	29.8269	2045.563	32.3513	2161.589	28.1710
	36-32-43	828.576	29.8154	1227.907	32.1184	1429.018	28.1512
	40-36-47	548.606	29.8079	798.278	31.8784	954.494	28.1013
9	28-24-37	2086.070	29.8029	3988.517	32.2862	3393.797	28.1091
	32-28-41	1221.288	29.8012	1986.101	32.2351	2128.368	28.0993
	36-32-45	797.275	29.8002	1197.254	32.0283	1410.744	28.0569
	40-36-49	530.952	29.7974	781.234	31.8232	944.894	27.9828

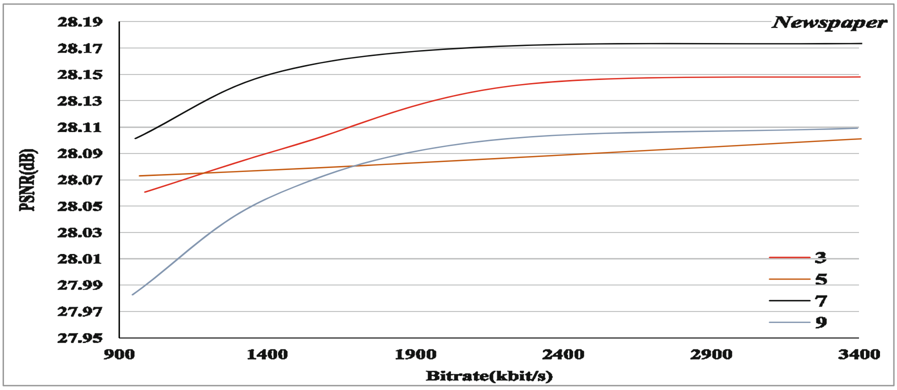
In Table 1, the *QP Difference* (*may be* 3, 5, 7, 9) means the difference of the QP between the BL and EL2. *Bitrates* is the summation of the bit rate of the texture video and the depth map when encoding them together. *SynView PSNR* is the PSNR of the synthetic view. According to the experimental results in Table 1, the RD curves can be drawn, as shown in Fig. 5.



(a)



(b)



(c)

Fig. 5. RD curves of different sequences with *QP Difference* = *n:* (a) *Ballet;* (b) *Breakdancers;* (c) *Newspaper.*

From Fig. 5, we can see that the synthetic view can obtain the best performance when encoding the *Newspaper* sequence with *QP Difference = 7*. For *Ballet* and *Breakdancers*, although when the higher PSNR is gained by using *QP Difference = 3*, the difference of PSNR between *QP Difference = 3* and *7* is very small. Besides, the bit rates of the sequences with *QP Difference = 7* are smaller than the sequences with *QP Difference = 3*, so the PSNR difference could be ignored.

Furthermore, the relationship of bit rates between texture videos and depth maps are illustrated in Table 2. From Table 2, we can see that the relationships of bit rates between texture videos and depth maps are basically consistent with [14] except a little difference when *QP Difference = 7*. This is because the *Ballet* sequence has the most violent motion, so it needs more bitrate to transmit the depth map, while the newspaper sequence which has the least motion needs less bitrate for the depth map. So, *QP Difference = 7* is chosen as the optimal in our experiments.

In addition, combined with the proposed BBS edge detection algorithm which is applied for mode decision process in EL2, the dynamic quantification for the depth map to improve its edge quality is applied. The flowchart of the BBS edge detection algorithm and dynamic quantification can be seen in Fig. 6.

According to the depth boundary effect on the quality of the synthetic view, BR needs to be preserved as much as possible with a smaller QP while HR can be quantified

Table 2. The relationship of bit rates between texture videos and depth maps

QP difference	QP	Bitrate (kbps)					
		Ballet		*Breakdancer*		*Newspaper*	
		Texture	Depth	Texture	Depth	Texture	Depth
3	28-24-31	1762.738	706.272	3653.592	768.590	3190.685	424.675
	32-28-35	1030.838	431.981	1785.014	448.243	1999.195	260.839
	36-32-39	680.525	258.331	1069.925	264.590	1324.349	162.245
	40-36-43	452.928	151.829	694.862	161.059	882.648	104.366
	Proportion	38.3 %		23.5 %		12.5 %	
5	28-24-33	1762.738	557.899	3653.592	586.315	3190.685	335.362
	32-28-37	1030.838	329.016	1785.014	339.638	1999.195	202.939
	36-32-41	680.525	193.877	1069.925	204.182	1324.349	128.890
	40-36-45	452.928	119.544	694.862	129.245	882.648	86.314
	Proportion	29.5 %		18.2 %		10.0 %	
7	28-24-35	1762.738	427.051	3653.592	442.387	3190.685	260.726
	32-28-39	1030.838	253.052	1785.014	339.638	1999.195	162.394
	36-32-43	680.525	148.046	1069.925	157.977	1324.349	104.669
	40-36-47	452.928	95.678	694.862	103.416	882.648	71.846
	Proportion	22.8 %		15.3 %		8.0 %	
9	28-24-37	1762.738	323.339	3653.592	334.925	3190.685	203.112
	32-28-41	1030.838	190.450	1785.014	201.086	1999.195	129.173
	36-32-45	680.525	116.750	1069.925	127.330	1324.349	86.395
	40-36-49	452.928	78.024	694.862	86.371	882.648	62.246
	Proportion	17.5 %		11.0 %		6.6 %	

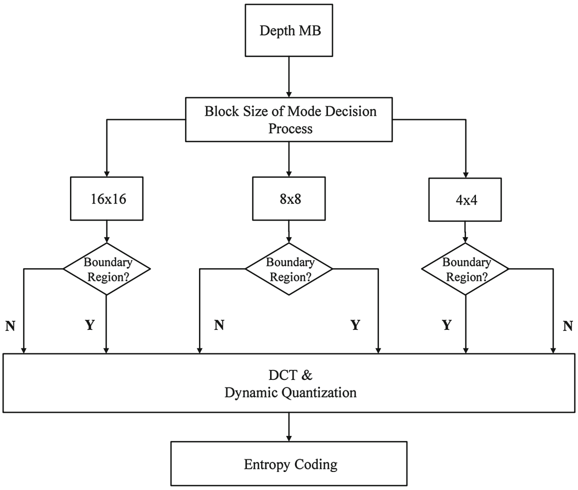


Fig. 6. The flowchart of the proposed BBS edge detection algorithm.

with a larger QP than the initial QP set in the configuration file. So a dynamic quantization based on the depth BR is proposed, and the final QP can be expressed as:

$$QP = QP_{init} + \Delta QP_{H/B} \tag{3}$$

where QP_{init} is the initial QP set in the configuration file and $\Delta QP_{H/B}$ is a variable value that can be adjusted based on block sizes and the region. When the current block belongs to BR, $\Delta QP_{H/B}$ is a negative value, otherwise it is a positive value. And the smaller the block size is, the more important the block will be. So it should be quantified with a small QP. In our experiments, $\Delta QP_{H/B}$ is set as:

$$\Delta QP_{H/B} = \begin{cases} -2, CurrentBlock \in BR \ and \ 16 \times 16 \\ -3, CurrentBlock \in BR \ and \ 8 \times 8 \\ -5, CurrentBlock \in BR \ and \ 4 \times 4 \\ 4, CurrentBlock \in HR \end{cases} \tag{4}$$

Since different QP are set for the boundary regions and homogeneous regions, the average changes of the amount of the bit stream will be small.

3 Simulation Results

Some experiments are carried out for evaluating the proposed method in SVC reference software JSVM 9–19. Three different video sequences: *Ballet, Breakdancers*, and *Newspaper*, are tested in this paper. The main encoding parameters are presented in Table 3.

The same inter-layer prediction is applied both in our coding scheme and the standard JSVM model. In addition, the initial QP for the depth map is selected by

Table 3. Main encoding parameters for evaluating the proposed algorithm.

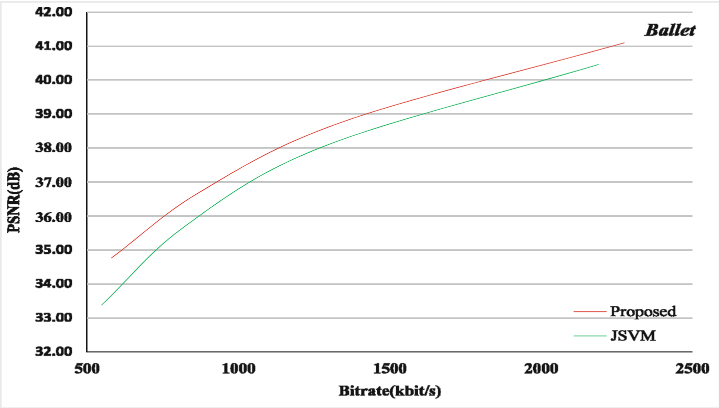
Sequences		*Ballet* (View2, V + D)
		Breakdancers (View4, V + D)
		Newspaper (View4, V + D)
Layer	BL and EL1	Texture Video
	EL2	Depth Map
Resolution		1024 × 768
QP (BL, EL1, EL2)		(28,24,35), (32,28,39)
		(36,32,43), (40,36,47)
Frame Rate		30 fps
GOP Size		8
Inter-layer Prediction for EL2		Inter-layer Motion Prediction: Yes
		Inter-layer Residual Prediction: No
		Inter-layer Intra Prediction: No
Frames to be coded		50

adding 7 to the QP for the texture video, which has been verified in the previous section. The bitrate and PSNR values of the proposed method and the standard three-layer JSVM model without edge detection are shown in Table 4.

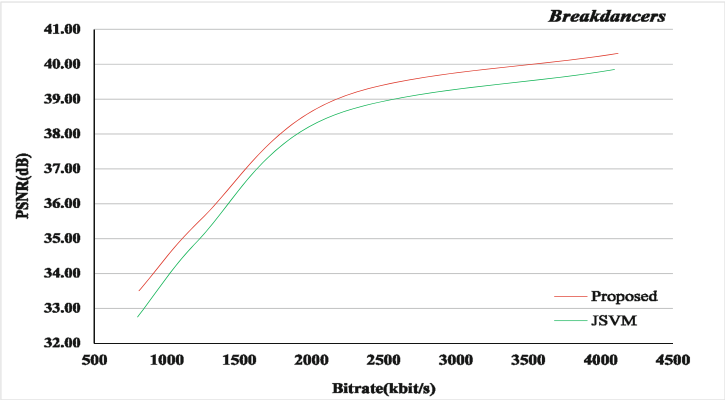
Since the texture video inputted to the BL and EL1 is processed in the same way with the texture video in JSVM standard model, the reconstructed quality of BL and EL1 in our coding method are substantially unchanged comparing with the JSVM standard model. So the bitrates and PSNR in Table 4 represent overall bitrates of all layers and the PSNR of the depth map respectively. The Bjøntegaard bitrates (BDBR, %) and the Bjøntegaard PSNR (BDPSNR, dB) [15] between the proposed method and the original JSVM method are calculated. From Table 4, the proposed method can

Table 4. Comparison of the proposed method and the original JSVM method.

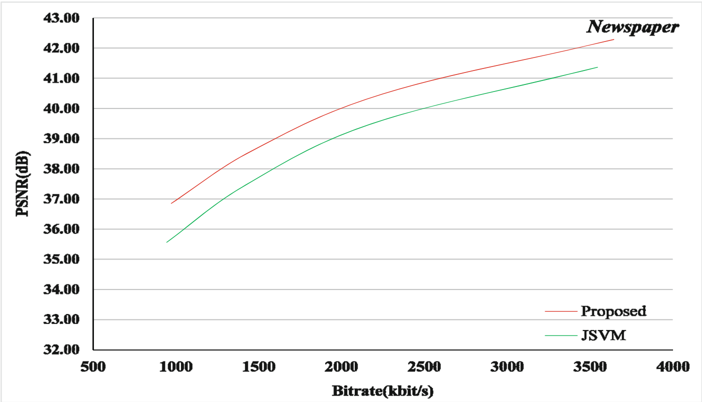
		Proposed		JSVM			
Sequences	QP	Bitrates (kbps)	PSNR (dB)	Bitrates (kbps)	PSNR (dB)	BDBR (%)	BDPSNR (dB)
Ballet	28-24-35	2275.070	41.097	2189.788	40.460	−11.52	0.616
	32-28-39	1348.286	38.783	1283.890	38.058		
	36-32-43	877.502	36.732	828.576	35.733		
	40-36-47	581.222	34.767	548.606	33.383		
Breakdancers	28-24-35	4120.349	40.311	4095.979	39.845	−9.72	0.463
	32-28-39	2185.906	39.020	2124.653	38.487		
	36-32-43	1246.968	35.591	1227.907	34.995		
	40-36-47	808.464	33.506	798.278	32.755		
Newspaper	28-24-35	3644.016	42.284	3545.774	41.366	−19.36	0.941
	32-28-39	2246.458	40.477	2186.299	39.502		
	36-32-43	1464.269	38.609	1421.616	37.461		
	40-36-47	972.624	36.852	945.667	35.563		



(a)



(b)



(c)

Fig. 7. RD performance comparisons: (a) *Ballet*; (b) *Breakdancers*; (c) *Newspaper*.

achieve the Bjøntegaard bitrates savings from 9.72 % to 19.36 % and the BDPSNR gains from 0.463 dB to 0.941 dB.

At last, the RD performance of the three sequences are depicted to evaluate the effectiveness of the proposed method as illustrated in Fig. 7. As we can see, our proposed method performs better than the standard JSVM model.

4 Conclusions

In our coding scheme, the texture video is encoded as BL and EL1 while its associated depth map is encoded as EL2. Considering the motion similarity between texture video and its associated depth map, inter-layer motion prediction is applied while the other two inter-layer prediction tools are not applied in the process of encoding the depth map. What's more, when selecting the optimal prediction mode, our proposed block-based Sobel edge detection algorithm is applied to judge whether the current block is in the boundary region or not. And then, the dynamic quantization is applied in boundary blocks and homogeneous blocks.

Simulation results show that the proposed method has an overall better performance than the standard JSVM model.

Acknowledgement. This work was supported by the National Natural Science Foundation of China under Grant (61301112).

References

1. Muller, K., Merkle, P., Tech, G.: 3D Video formats and coding methods. In: 2010 17th IEEE International Conference on Image Processing (ICIP), pp. 2389–2392. IEEE (2010)
2. Muller, K., Merkle, P., Wiegand, T.: 3-D video representation using depth maps. Proc. IEEE **99**(4), 643–656 (2011)
3. Wang, L., Hou, C., Lei, J., Yan, W.: View generation with DIBR for 3D display system. Multimedia Tools Appl. 1–17 (2014)
4. Wiegand, T., Sullivan, G.J., Bjontegaard, G., Luthra, A.: Overview of the H. 264/AVC video coding standard. IEEE Trans. Circ. Syst. Video Technol. **13**(7), 560–576 (2003)
5. Schwarz, H., Marpe, D., Wiegand, T.: Overview of the scalable video coding extension of the H. 264/AVC standard. IEEE Trans. Circ. Syst. Video Technol. **17**(9), 1103–1120 (2007)
6. Tao, S., Chen, Y., Hannuksela, M.M.: Joint texture and depth map video coding based on the scalable extension of H. 264/AVC. In: IEEE International Symposium on Circuits and Systems, ISCAS 2009, pp. 2353–2356 (2009)
7. Lei, J., Li, S., Zhu, C.: Depth coding based on depth-texture motion and structure similarities. IEEE Trans. Circ. Syst. Video Technol. **25**(2), 275–286 (2014)
8. Oh, H., Ho, Y.-S.: H.264-based depth map sequence coding using motion information of corresponding texture video. In: Chang, L.-W., Lie, W.-N. (eds.) PSIVT 2006. LNCS, vol. 4319, pp. 898–907. Springer, Heidelberg (2006)
9. Daribo, I., Tillier, C., Pesquet-Popescu, B.: Motion vector sharing and bitrate allocation for 3D video plus depth coding. EURASIP J. Appl. Sig. Process. (3), 3 (2009)

10. Zamarin, M., Salmistraro, M., Forchhammer, S.: Edge-preserving intra depth coding based on context-coding and H. 264/AVC. In: 2013 IEEE International Conference on Multimedia and Expo (ICME), pp. 1–6. IEEE (2013)
11. Kao, J., Tu, J.: The compression method of depth map in 3D video. In: 5th IET International Conference on Wireless, Mobile and Multimedia Networks (ICWMMN 2013), pp. 189–192. IET (2013)
12. Zhang, X., Xiong, H.: Multiscale edge coding and adaptive lifting for depth maps coding in 3-D video. In: Data Compression Conference (DCC), p. 439. IEEE (2014)
13. Liu, S., Lai, P., Tian, D., Chen, C.W.: New depth coding techniques with utilization of corresponding video. IEEE Trans. Broadcast. **57**(2), 551–561 (2011)
14. Smolic, A., Mueller, K., Stefanoski, N., Ostermann, J., Gotchev, A., Akar, G.B., Koz, A.: Coding algorithms for 3DTV—a survey. IEEE Trans. Circ. Syst. Video Technol. **17**(11), 1606–1621 (2007)
15. Bjøntegaard, G.: Calculation of average PSNR differences between RD-curves. Document VCEG-M33 ITU-T Q6/16, TX, USA, pp. 2–4. Austin (2001)