Bit-rate allocation for Multi-view video plus depth

Emilie Bosc, Vincent Jantet, Muriel Pressigout, Luce Morin, Christine Guillemot

BIT-RATE ALLOCATION FOR MULTI-VIEW VIDEO PLUS DEPTH

Emilie Bosc[1]*, Vincent Jantet*[2]*, Muriel Pressigout*[1]*, Luce Morin*[1]*, Christine Guillemot*[2]

[1] IETR - INSA Rennes - 20 avenue des Buttes de Coësmes - 35043 Rennes, France
[2] INRIA Rennes, Bretagne Atlantique - Campus de Beaulieu - 35042 Rennes, France

ABSTRACT

The efficient compression of multi-view-video-plus-depth (MVD) data raises the bit-rate allocation issue for the compression of texture and depth data. This question has not been solved yet because not all surveys reckon on a shared framework. This paper studies the impact of bit-rate allocation for texture and depth data relying on the quality of an intermediate synthesized view. The results show that depending on the acquisition configuration, the synthesized views require a different ratio between the depth and texture bit-rate: between 40% and 60% of the total bit-rate should be allocated to depth.

Index Terms— 3DTV, MVD, 3D video, MVC, quality assessment.

1. INTRODUCTION

Multi-view video plus depth (MVD) data is a set of multiple sequences capturing the same scene at different viewpoints, with their associated per-pixel depth value. MVD data lead to two main applications: 3D television (3DTV) that provides a depth feeling, and Free Viewpoint Video (FVV), that allows navigation inside the scene. Depth data provide information on scene geometry and help in virtual intermediate view generation. Since texture and depth information are required for view synthesis, an efficient coding framework should ensure the preservation of essential data. Indeed, previous studies ([1],[2]) have shown that coding artifacts on depth data can dramatically influence the quality of the synthesized view. Depth maps are not natural images. Most of the state-of-the-art used codecs for depth maps are inspired from 2D video codecs that are optimized for human visual perception of color images. Yet, a straightforward idea suggests that being a monochrome signal, depth maps require low bit-rate compared to texture data. Actually, because of its capital role in the view synthesis processing, compression artifact of such data may lead to fatal synthesis errors when generating virtual views. Consequently, a simple but essential question refers to the bit allocation ratio between texture and depth. This rate ratio depends on the targeted application. Here, we address this question by measuring the Peak Signal to Noise Ratio (PSNR) scores of intermediate views which need to be generated in contexts of 3DTV (for rendering on autostereoscopic displays) or of FVV for rendering view points different from those captured by the cameras. The use of this objective metric is justified by its simplicity and mathematical easiness to deal with such purposes, although previous studies [3] highlighted the need for new metrics for 3D video. The appropriate rate ratio that should be used is not clearly stated in the literature: most of the studies do not rely on the same framework. Fehn et al. [4] show that being a gray-scale signal, the depth video can be compressed more efficiently than the texture video using less than 20% of the texture bit-rate for video-plus-depth data format. In [5], the authors proposed an efficient joint texture/depth rate allocation method based on a view synthesis model distortion, for the compression of MVD data. According the the bandwidth constraints, the method delivers the best quantization parameters combination for depth/texture sets that maximizes the rendering quality of a synthesized view in terms of MSE.
Our experiments tried to quantify the appropriate rate ratio between depth and texture data, and then analyze the relationship with the encoded sequence. Section 2 is devoted to virtual view synthesis. Section 3 states the experimental protocol used to assess the influence of the texture/depth compression ratio while Section 4 discusses the results. Finally Section 5 concludes the paper.

2. VIRTUAL VIEW SYNTHESIS

For 3DTV or FVV, the transmitted texture and depth sequences are used to generate virtual views with the help of depth-image-based rendering techniques. The generated views can then be rendered on a conventional display, or a stereoscopic or an autosterescopic display.
Generating a "virtual" view consists in synthesizing a novel view of the scene, from a viewpoint which differs from those captured by the cameras, relying on the available texture and depth data. The texture, that is the conventional 2D color sequences, gives the color information. The depth data are gray-scales images and are considered as a monochromatic signal. Each pixel of a depth image, also called depth map, indicates the distance of the corresponding 3D-point from the camera. Based on projective geometry [6], the 3D representation of a scene can be retrieved from a depth map. The presented experiments are based on the View Synthesis

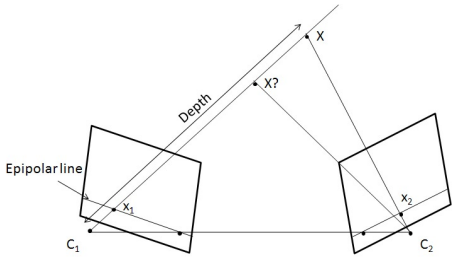


Fig. 1. Relationship between image points and real world [8].

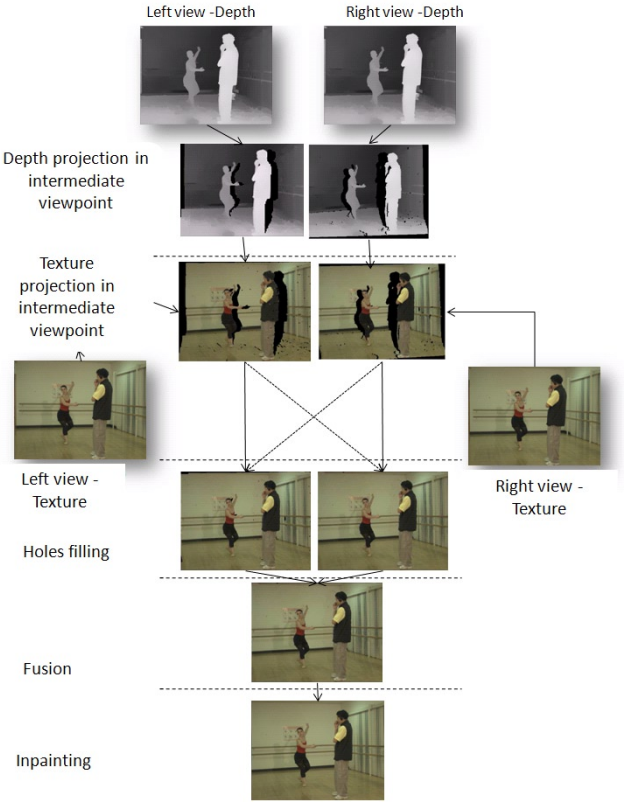


Fig. 2. View synthesis with VSRS.

Reference Software (VSRS) [7], version 3.5, provided by MPEG. Figure 1 illustrates the relationship between a real 3D point of the scene, defined as X and its projections x_1 and x_2 in camera planes of C_1 and C_2 respectively. The geometric transformation from 3D world to the camera plane can be easily performed from depth data, and both intrinsic and extrinsic parameters of known cameras. On the same principle, 3D points of the real world can be projected onto the image plane of a virtual camera from an arbitrary viewpoint. VSRS is able to deal with such cases, providing it the parameters related to the virtual camera C_v and texture and depth information from two adjacent known views. Figure 2 shows the principles of synthesis used in VSRS. Depth maps of the two adjacent views are projected into the target view resulting in two new depth maps referring to the virtual viewpoint. Those new maps contain non-valued areas, called holes. They correspond to occluded areas in the reference viewpoint (left or right respectively). Consequently, when the left and right texture images are projected in the target viewpoint according to the new depth maps, they also contain non-valued areas. Those areas are then filled in by available information from both new texture images. Then the two texture images are fused into one single image that is usually post-processed by in-painting methods.

The described process assigns each pixel of the new texture image T_v, a color value according to its corresponding depth. Three cases are considered:

- both depth values for the considered pixel are null: this is an non visible area.
- only one of the two pixels has a depth value: this is a occluded area in one of the reference viewpoints.
- depth values of the pixels in the adjacent views are not null. This is expressed by:

$$T_v = \begin{cases} 0, & \text{if } (u,v) \text{ is not visible} \\ T_1(u,v), & \text{if } d_1(u,v) \neq 0 \text{ and } d_2(u,v) = 0 \\ T_2(u,v), & \text{if } d_1(u,v) = 0 \text{ and } d_2(u,v) \neq 0 \\ (1-\alpha)T_1(u,v) + \alpha T_2(u,v), & \text{if } d_1(u,v) \neq 0 \text{ and } d_2(u,v) \neq 0 \end{cases}$$

where (u, v) refers to the coordinates of a pixel of the synthesized view, $d_1(u, v)$ is the depth value of this pixel from camera C_1, $d_2(u, v)$ is the depth value of this pixel from camera C_2, and α is a factor depending on the distance ($\alpha < 1$). The synthesis process already raises some issues. First, in terms of geometry: occluded regions lead to non-valued areas in the texture image. Secondly, errors can occur because pixel coordinates do not locate at an integer position and is usually either interpolated or rounded to the nearest integer position. In-painting methods as well as interpolation filters are developed in order to reduce the synthesis artifacts.

3. PROTOCOL

Our aim was to evaluate the required ratio between depth and texture data relying on the quality of a reconstructed view, in terms of PSNR. Thus, we used the Multiview Video Coding (MVC) reference software, JMVM 8.0 (Joint Multiview Video Model) to encode three views, as a realistic simulation of a 3DTV use. To vary the bit-rate ratio and the total bit-rate, the quantization parameter QP varies from 20 to 44 for both depth and texture coding. The central view predicts the two other views. Then, from the decompressed views, we computed the intermediate view between the central view and the right one, by using the reference software: VSRS, version 3.5, provided by MPEG.

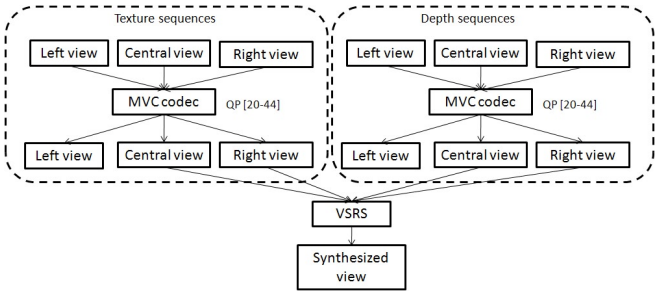
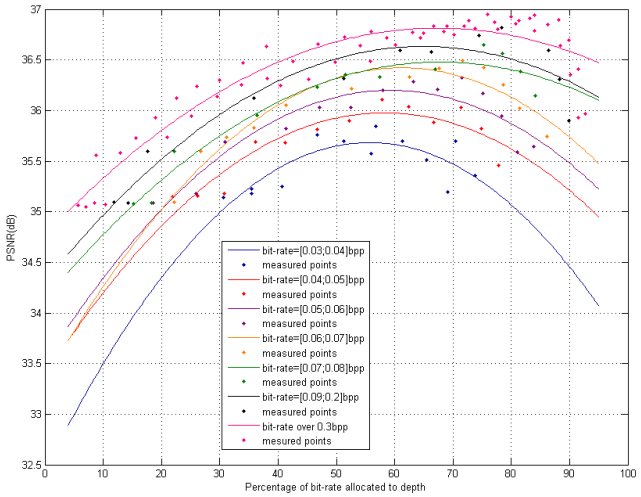


Fig. 3. Experimental protocol.

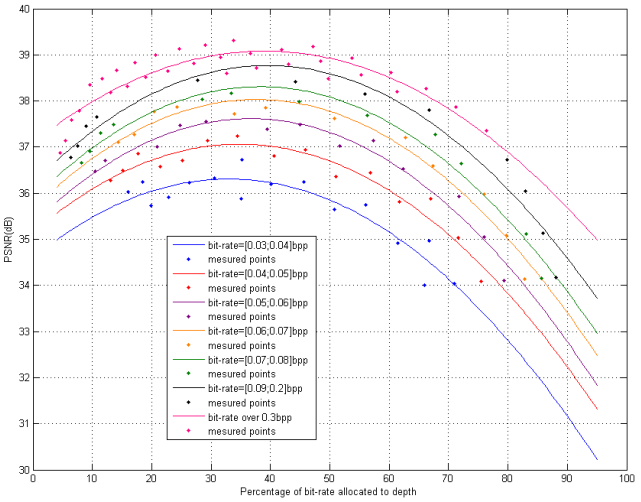
Figure 3 illustrates the described protocol. We used two different types of sequences to answer our question: *Ballet* from Microsoft Research, and *Book Arrival* from Fraunhoffer HHI (1024 $\times$ 768). This last sequence is a 3DV test material in MPEG. It was acquired through cameras arranged equidistantly along a straight line with a rectified configuration (no gap in-between the cameras and the interaxial distance is 65 mm). On the other hand, *Ballet* was acquired with converging cameras. The encoded views are 2, 4 and 6 for *Ballet*, and 6, 8 and 10 for *Book Arrival*. For each couple $(QP_{texture}, QP_{depth})$, the average PSNR score of the synthesized sequence is evaluated, compared to the original acquired view.

4. RESULTS AND DISCUSSION

Figure 4 presents the results. The average PSNR of the synthesized sequence are plotted over the bit-rate percentage assigned to depth. The different curves correspond to interpolation of the measured points for different ranges of bit-rate. We observe that for a given sequence, no matter the bit-rate, the ratio that provides the best quality is the same: it seems to be around 60% for *Ballet*, and around 40% for *Book Arrival*. This suggests that the required depth information that enables a good reconstruction quality, in terms of PSNR depends on the content. More precisely, we think that this percentage depends on the acquisition configuration of the sequence, i.e. the camera baseline. The synthesized sequence from *Book Arrival* may require less elements of depth for the reconstruction because of the linear configuration of the three used cameras. On the contrary, the synthesized sequence from *Ballet*, seems to require an important amount of depth information to ensure a good quality of reconstruction. Correct reconstruction around dis-occluded areas may require more reliable depth information depending on the reference camera position. These results are consistent with [5]: although the authors do not state clearly the appropriate ratio for those two sequences, their rate/distortion curves show that, for example, the bit-rate pair (962kbps for texture, 647kbps for depth), i.e. a percentage of 40% for depth, gives better synthesis quality (in terms of PSNR) for *Book Arrival*. The synthesis conditions are similar to our experiments. On the other hand, in [4], the synthesis conditions involves video-plus-depth data: in this case, a very little continuum is supported around the available original view. Since synthesis distortion increases with the distance of the virtual viewpoint, this explains the significant difference with our results.



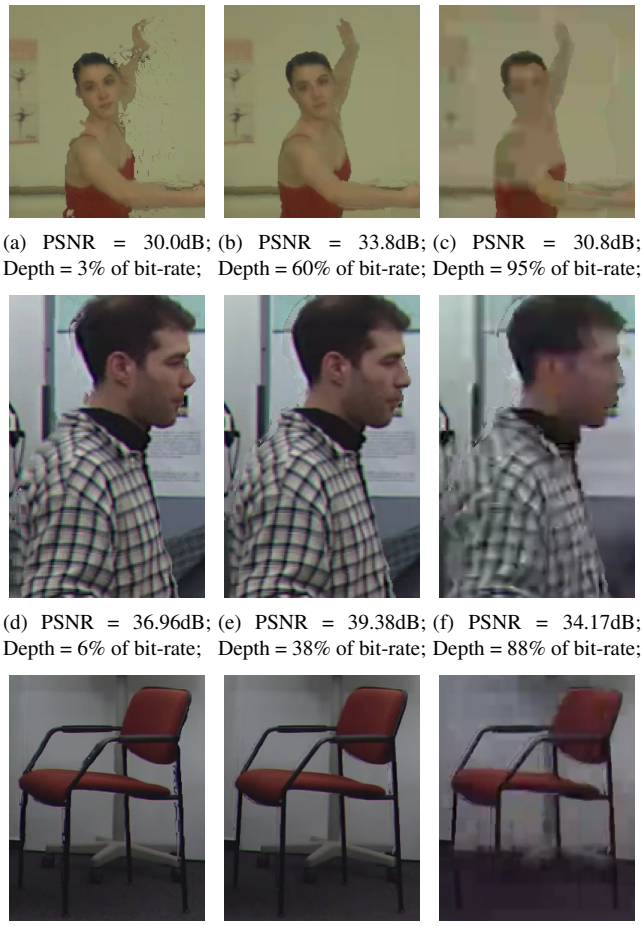
(a) PSNR (dB) of synthesized views as a function of rate allocated to depth in percentage of total rate for Ballet



(b) PSNR (dB) of synthesized views as a function of rate allocated to depth in percentage of total rate for Book Arrival

Fig. 4. Interpolated rate-distortion curves of synthesized views.

Figure 5 shows particular areas of the synthesized views from the presented experiment. Figures 5(a), 5(d) and 5(g) show that allocating less than 10% of the total bit-rate to depth data induces important damages along the edges. The location of the depth map discontinuities is deeply compromised which leads to errors in the synthesized view. PSNR scores



(a) PSNR = 30.0dB; Depth = 3% of bit-rate; (b) PSNR = 33.8dB; Depth = 60% of bit-rate; (c) PSNR = 30.8dB; Depth = 95% of bit-rate;

(d) PSNR = 36.96dB; Depth = 6% of bit-rate; (e) PSNR = 39.38dB; Depth = 38% of bit-rate; (f) PSNR = 34.17dB; Depth = 88% of bit-rate;

(g) PSNR = 36.96dB; Depth = 6% of bit-rate; (h) PSNR = 39.38dB; Depth = 38% of bit-rate; (i) PSNR = 34.17dB; Depth = 88% of bit-rate;

Fig. 5. Synthesized images from MVD data, with different bit-rate ratios between texture and depth.

fall down because of the numerous errors along the contours of objects. Figures 5(c), 5(f) and 5(i) suggest that affecting more than 80% of the total bit-rate to depth data preserves the edges of some objects but texture information is lost because of the coarse quantization. Affecting between 40% and 60% of the total bit-rate to depth data seems to be a good trade-off for the tested sequences, as it can be observed in Figure 5(b), 5(e) and 5(h). PSNR scores and visual quality are both improved compared with the two other presented cases. The depth maps are accurate enough to ensure correct projections and decompressed texture images are good enough to avoid drastic artifacts.

5. CONCLUSION

This paper aimed at determining the appropriate ratio for joint depth/texture compression, in the MVD framework. The experiment consisted in encoding both texture and depth data by the same compression scheme, that is to say using the MVC coder. The attributed depth ratio was varied from 2% to nearly 95% and the synthesis of an intermediate view was performed. The average PSNR score of the synthesized views was then evaluated. As expected, the compression of texture data induces blurred regions in the synthesized views, while the coarse compression of depth data leads to geometrical errors during the pixels projection. An ideal ratio between depth and texture can be found in order to maximize the quality of the synthesized view; this ratio seems to be the same for a given sequence no matter the total rate. The obtained results showed that the best quality of reconstruction by using VSRS may require to affect between 40% and 60% of the total bit-rate to depth data, depending on the available MVD data. Those observations are related to H.264/MVC encoding. Using a different encoding framework may allow to reduce the ratio for depth. In future work, the analysis of the MVD data and related parameters such as video contents and camera settings should be studied to automatically set the correct ratio between depth and texture according to these parameters.

6. ACKNOWLEDGMENTS

This work is supported by the French National Research Agency as part of PERSEE project (ANR-09-BLAN-0170) and CAIMAN project (ANR-08-VERS-002). We would like to acknowledge the Interactive Visual Media Group of Microsoft Research for providing the "*Ballet*" data set, Fraunofer HHI for providing the "*Book Arrival*" data set and MPEG for providing VSRS.

7. REFERENCES

[1] P. Merkle, A. Smolic, K. Muller, and T. Wiegand, "Multi-view video plus depth representation and coding," in *Proceedings of ICIP*, 2007, p. 201204.

[2] A. Vetro, S. Yea, and A. Smolic, "Towards a 3D video format for auto-stereoscopic displays," *Proceedings of the SPIE: Applications of Digital Image Processing XXXI, San Diego, CA, USA*, 2008.

[3] A. Tikanmaki, A. Gotchev, A. Smolic, and K. M\ller, "Quality assessment of 3D video in rate allocation experiments," in *IEEE Int. Symposium on Consumer Electronics (14-16 April, Algarve, Portugal)*, 2008.

[4] C. Fehn et al., "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 XI*, 2004, vol. 5291, p. 93104.

[5] Y. Liu, Q. Huang, S. Ma, D. Zhao, and W. Gao, "Joint video/depth rate allocation for 3D video coding based on view synthesis distortion model," *Signal Processing: Image Communication*, vol. 24, no. 8, pp. 666–681, Sept. 2009.

[6] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*, Cambridge Univ Pr, 2003.

[7] M. Tanimoto, T. Fujii, K. Suzuki, N. Fukushima, and Y. Mori, "Reference softwares for depth estimation and view synthesis," Apr. 2008.

[8] C. Lee and Y. S Ho, "View synthesis tools for 3D Video.ISO/IEC JTC1/SC29/WG11 MPEG2008/M15851," Oct. 2008.