Fast View Synthesis using platelet-based depth representation

Krzysztof Wegner, Olgierd Stankiewicz, Marek Domański
Chair of Multimedia Telecommunications and Microelectronics
Poznań University of Technology
Polanka 3, 61-131 Poznań, Poland
kwegner@multimedia.edu.pl

***Abstract* - This paper presents a novel approach to speeding-up the view synthesis process. A general case of view arrangement in a 3D video system is considered. The proposed approach offers a certain acceleration practically without any quality loss, and further acceleration at the cost of lower quality of synthesis. Amount of the acceleration can be gradually exchanged with the degradation of synthesized image quality via adaptive block size depth data partitioning. The proposal exploits depth modeling - the depth data is divided into blocks of various sizes and modeled as planes. In such case only 4 corners of each block need to be transformed during view synthesis, instead of transforming every pixel in the block. This way depth data is adaptively simplified. The experimental results on wide range of multiview test sequences are provided, for both original and compressed depth data.**

Keywords **– View Synthesis; DIBR; Free view television; Depth maps; Platelet**

I. INTRODUCTION

Recent years have yielded rapid development of technologies related to a new generation of 3D video systems. Services like glasses-free viewing with use of autostereoscopic displays or free-view point navigation controled by the user employ synthesis of virtual views in order to overcome the limitations of classical video. Without view synthesis, the desired number of views not only could not be delivered – due to enormous bandwidth consumption - but also would be even impossible to acquire, because very dense placement of cameras is unattainable due to physical dimensions of the camera bodies. View synthesis is expected to be realized by user-side devices, like 3D displays, mobile phones, etc. Therefore there is a need for fast, computationally efficient view synthesis algorithms, which desirably could be seamlessly implemented in hardware.

Currently, the most common representation of a scene is MultiView-plus-Depth (MVD) [1], in which information about three-dimensional structure of a scene is stored in form of depth maps. Basing on depth maps and corresponding videos (so called texture views), a virtual view can be synthesized, typically in between of the source views. The use of MVD leads to the simplest and straightforward approach to view synthesis, which is to employ Depth-Image-Based Rendering (DIBR) [2]. The most general idea of DIBR says that pixels $m_s = [u \quad v \quad 1]^T$ of the source views are projected with use of the projection matrix P_s from 2D-image plane to 3D-space $M = [X \quad Y \quad Z]^T$, and then projected back to pixel m_v onto image plane of the virtual view with use of projection matrix P_v (1).

$$\begin{matrix} M = P_s^{-1} \cdot m_s \\ m_v = P_v \cdot M \end{matrix} \Rightarrow m_v = P_v P_s^{-1} \cdot m_s \tag{1}$$

In the references [3-5], several variants of DIBR-based view synthesis have been described. These variants differ by small improvements in pre- or post-processing. A very important and interesting use-case corresponds to the linear arrangement of cameras that is related to applications with autostereoscopic displays. In such a case, a combined Homography transformation 2D-3D-2D (2) is simplified to a simple, pure translation of points, which can be efficiently implemented as a LookUp-Table (LUT).

$$H_{vs} = P_v P_s^{-1} \tag{2}$$

There is lack of research in field of computational optimization of other cases with arbitrarily arranged cameras (i.e. arc arrangement). In work [3], the only optimization applied is to use a set of Homography transforms H_{vs} reduced to 3x3 matric (3x3 matrix multiplication), one per distance plane.

The state-of-the-art in DIBR-like view synthesis is represented by View Synthesis Reference Software (VSRS) [5] developed by ISO/IEC MPEG group of International Standardization Organization, in the course of works on 3D television standardization. The technique implemented in VSRS (like many others with some slight variants) consists of the following steps:

1) Forward Depth Warping – where pixels of depth map are projected from source view into the target view (warped forward). This step incorporates z-test, in which closer pixels occlude further pixels (simplification of this step is considered in the paper).
2) Depth Post-processing – in this step, newly generated target depth map is processed. This includes filing of small holes which has not been covered by any of warped source pixels due to warping. Often this is done by a mean of median filtering. Large, disoccluded regions are not filled in this step.
3) Backward Texture Warping – basing on the post-processed target depth map generated in step 2, texture of the source view is warped. This step is simpler than step 1, as it only involves addressing texture pixels.

4) Image Merging – the abovementioned operations (steps 1 to 3) are performed for each of the source views. This results with alternate versions of the target synthesized view. In this step, the contents of those alternate versions are merged, typically by averaging, and holes are filled with the content of the alternate version of the target view.
5) Hole Inpainting - finally, as still some parts of the target view may be unavailable, because are occluded in all of the source views, texture inpainting is performed. [8].

The abovementioned approach is often considered for implementation of Graphical Processing Units (GPU) which are multi-scalar processor arrays incorporated in modern graphics cards for PC computers and also in mobile devices. In such approach, the most expensive step of the view synthesis process is Forward Depth Warping. The most commonly, it is implemented as multiplication of 4x4 homography matrix per 4-element pixel position vector. This is performed for every pixel in the image, which is very expensive. In this work, we propose a novel approach to view synthesis, in which this most expensive step, Forward Depth Warping, is dramatically lightened.

II. THE IDEA OF THE PROPOSAL

The main idea of our proposal lies in simplification of the representation of the depth data. Instead of dense, regularly sampled pixels in depth map, we propose to represent depth in a similar way as in case of platetet-based compression technique [9]. In our approach rectangular blocks of depth pixels are modeled with flat planes, described by four corners. In this case complexity of view synthesis process is significantly reduced. Instead of pixel by pixel transformation, plane-model-based transformation can be used (Fig. 1).

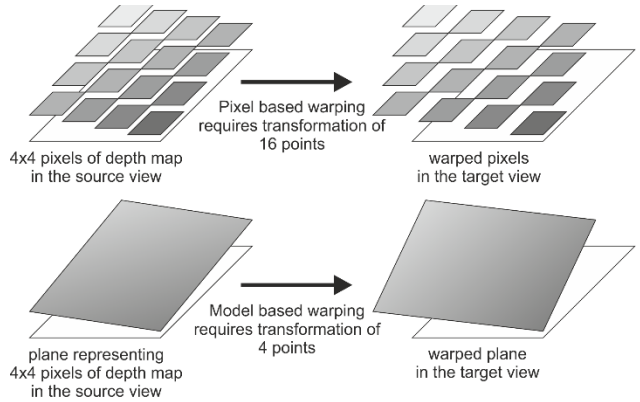

Figure 1. The main idea of the proposed method.

Therefore, in our method, projection of the given block of size $N \times N$ pixels can be done by projection of only 4 corners of the plane model, instead of N^2 pixels. The theoretical $acceleration$ of projection can thus be described as:

$$acceleration = \frac{N^2}{4} \tag{3}$$

This theoretical acceleration is of moderate practical use, as it describes speeding-up of projection of a single block of depth only and does not provide any indication about degradation of the quality of the modeled depth. Therefore, we compared the quality obtained with the proposed method to commonly used view synthesis approach.

III. DEPTH MAP BLOCK MODEL

The depth for a given block is modeled as a plane defined by equation:

$$z'(x,y) = A \cdot x + B \cdot y + C \tag{4}$$

where: $z'(x,y)$ is depth value for coordinates (x,y) of the planar-depth model in a given block, A, B, C are parameters of an arbitrary plane, constants for a given block.

For a given block, plane parameters A, B and C can be estimated basing on actual content of the modeled depth map $z(x,y)$, with use of least-square energy minimization. We assume that we search for such a set of parameters that minimizes energy of $error$ in a given block of size $N \times N$:

$$error = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \left(z'(x,y) - z(x,y)\right)^2 \tag{5}$$

It can be derived that:

$$\begin{aligned} A &= \frac{12S_{xz} - 6S_z(N-1)}{N^2(N+1)(N-1)} \quad , \quad S_{xz} = \sum_{x=0}^{N-1} x \sum_{y=0}^{N-1} z(x,y) \quad , \\ B &= \frac{12S_{yz} - 6S_z(N-1)}{N^2(N+1)(N-1)} \quad , \quad S_{yz} = \sum_{y=0}^{N-1} y \sum_{x=0}^{N-1} z(x,y) \\ C &= \frac{-6S_{xz} - 6S_{yz} + S_z(7N-5)}{N^2(N+1)} \quad , \quad S_z = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} z(x,y) \quad , \end{aligned} \tag{6}$$

where $z(x,y)$ is the depth map value for coordinates (x,y).

IV. DEPTH MAP PARTITION

The proposed idea can be applied in various ways. In particular, the partitioning of depth image into square blocks can be done very arbitrarily. We have decided to model the depth data in a content adaptive manner. We have divided the depth data into the blocks basing on the Lagrangian optimization, which choose between depth representation accuracy (model *error* – depth quality) (5) and the amount of the *acceleration* (3). Lagrangian multiplier λ allow control of ratio between depth accuracy and the speed-up-ratio:

$$fitness = \lambda \cdot error + acceleration \tag{7}$$

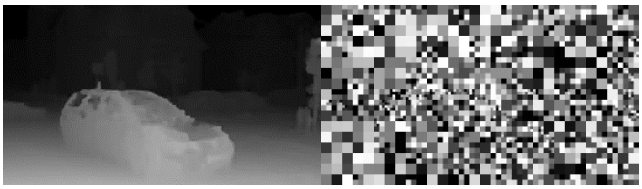

Figure 2. An exemplary depth frame of Poznan Street sequence [11], partitioned into a blocks of various size (right) and modeled (left). Partitioning has been illustrated as blocks uniformly shaded with random gray level.

Each of the depth images (for each view separately) is partitioned in a quad-tree based manner. For sake of practicality, we consider block sizes from 1×1 (single pixel) up to 64×64 pixels. Such a range of block sizes align well with state-of-the-art compression technology (and thus video representation) trends [9, 10].

V. EVALUATION

We have evaluated our approach with use of commonly known 3D video sequences, widespread by ISO/IEC MPEG group during announcement of "Call for Proposals for 3D Video Coding Technology" [12]. Those sequences consist of multiview high-resolution video data along with high quality depth maps and are provided for scientific purposes (Fig. 3).

For each of the used test sequences, depth maps from three input views have been modeled with use of the proposed technique. The resultant depth models for those views , along with corresponding textures, have been used to synthesize virtual views in 6 intermediate positions, equally spaced between the input views. Average quality of those 6 synthesized views, measured as a luminance PSNR with respect to views synthesized using the original unprocessed depth data has been evaluated.

Figure 3. Test sequences used in evaluation, from top-left: Poznan Street, Poznan Hall, Poznan Carpark [11], Newspaper [13], Undo Dancer [14], GT Fly [15], Kendo, Balloons [16].

The experiments have been performed on original depth data (uncompressed), as well as on compressed depth data in order to evaluate the acceleration obtainable in a real world scenarios, where the quality of depth is degraded during transmission. Videos and depth data was compressed with use of a state-of-the-art 3D video coding technology, namely 3D-HEVC [10] developed by Joint 3D Video Coding Team (JCT-3V) of MPEG and VCEG. As for compression, 4 different rate-points have been used, recommended by JVT-3V for compression performance evaluations [12]. In particular QP/QD pairs of (25,34), (30,39), (35,42), (40,45) have been used. The experiments have been performed for a wide range of Lagrangian multipliers, which resulted in comprehensive "*quality versus acceleration*" curves.

VI. RESULTS

Fig. 4 shows the degradation of PSNR in a case, when depth modeled with the proposed method is used for view synthesis instead of original data. For most sequences, the plateau of the curve is reached at about 60dB, which corresponds to almost no visible degradation. This corresponds to different speed-up ratios, depending on the sequence: from about ×2 (for GT_Fly) to about ×12 (for Poznan Hall 2). It can be noticed that for acceleration from ×32 to ×512, degradation of PSNR is no lower than 40dB versus the original data.

Fig. 5 show similar comparisons but with respect to depth data compressed with 3D-HEVC. Here, we consider already degraded quality and thus some of modeling artifacts are hidden, which allows higher acceleration. For example of QP/QD pair (30,39), acceleration of ×16 leads to no further degradation of quality, related to compressed (but not modeled) depth data. Further acceleration to ×512, leads to degradation to PSNR level of about 42dB for most of sequences, with exception of Dancer and Newspaper sequences, where such acceleration leads to degradation to a level of about 35dB.

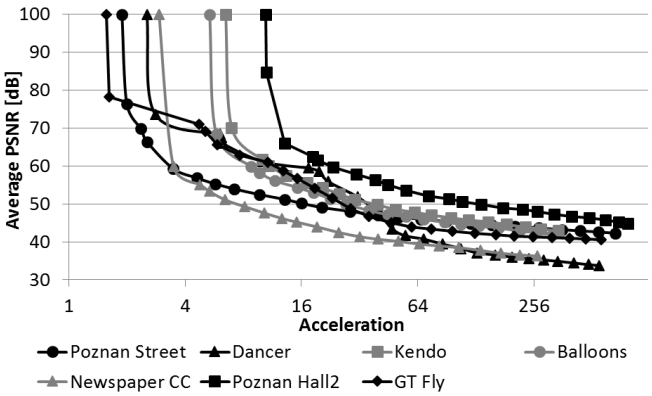

Figure 4. Depth modeling of the original uncompressed data. PSNR averaged over the synthesized views (measured with respect to the original uncompressed data case) versus acceleration.

In Fig. 6, the same phenomenon is shown as an average over the sequence set. It can be seen that for QP/QD pair (25,34), there is almost no degradation penalty for acceleration of about ×16, while for acceleration of ×256, the quality lowers to about 42dB. On the other hand, for QP/QD pair (40,45), there is very slight degradation from about 41.5dB to 40dB, while acceleration ranges from ×6 to ×256.

The loss of PSNR, averaged over all of the sequences is presented in Fig. 7. It can be notices, that for acceleration of ×256, quality degraded proportionally to the quantization level used for compression – the higher compression (higher QP), the lower the delta between the quality of the views synthesized based on modeled and not modeled (but still compressed) depth data.

VII. CONSLUSSIONS

In this paper, a novel approach for view synthesis acceleration based on depth map simplification has been presented. As it has been shown both theoretically and empirically, the proposal leads to a vast rendering speeding-up, within the range from×16 to ×256 with still acceptable level of synthesized image quality degradation. The level of synthesis quality degradation and resulting projection acceleration can be controlled by a constant used in Lagrangian optimization of presented depth modeling. The proposed tool of depth map modeling and simplification is similar to a well-known platelet based depth map representation [9].

VIII. ACKNOWLEDGMENTS

Research project was supported by National Science Centre, Poland according to the decision DEC-2012/05/N/ST6/03378.

Olgierd Stankiewicz and Krzysztof Wegner are scholarship holders of Scholarship support for Ph.D. students specializing in majors strategic for Wielkopolska's development, Sub-measure 8.2.2 Human Capital Operational Programme, co-financed by European Union under the European Social Fund.

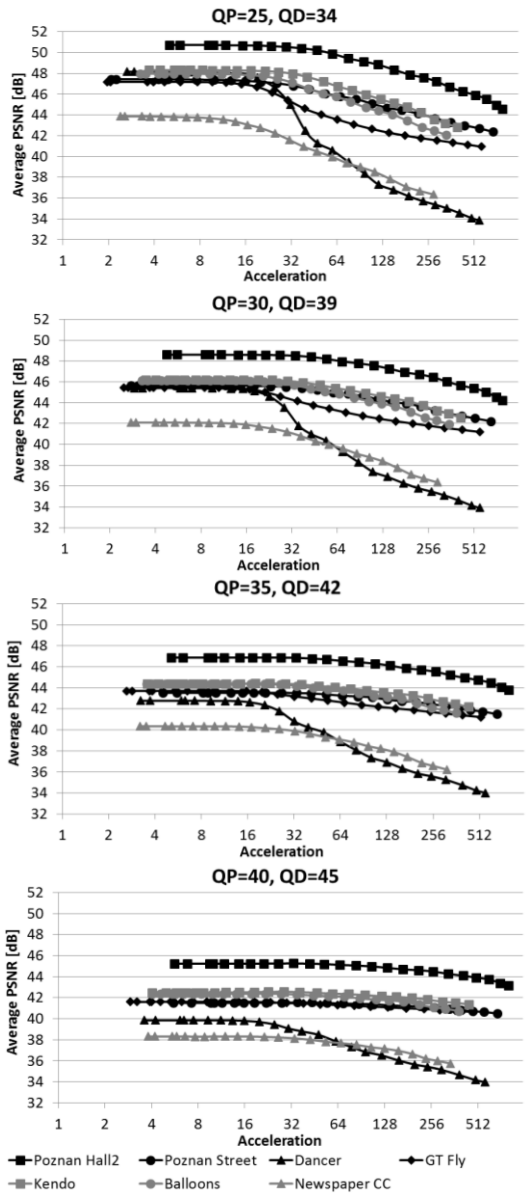

Figure 5. Average view synthesis quality using videos and depths compresed by 3D-HEVC with various QP pairs versus acceleration.

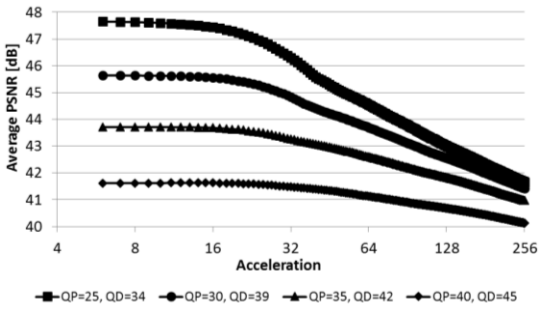

Figure 6. Comparison of view synthesis quality in terms of PSNR averaged over all of the sequences versus acceleration.

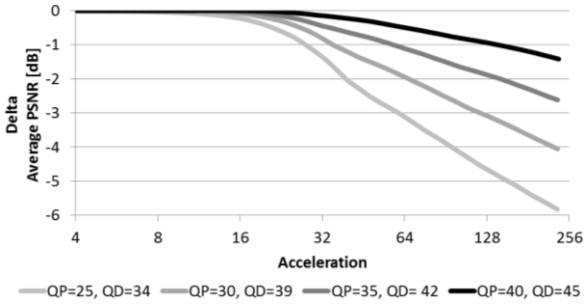

Figure 7. Comparison of average view synthesis quality lose caused by projection acceleration versus acceleration for view synthesis based on compresed data with different QPs.

IX. REFERENCES

[1] P. Merkle, A. Smolic, K. Muller, T. Wiegand, "Multi-View Video Plus Depth Representation and Coding," Image Processing, 2007. ICIP 2007. Sept. 16 2007-Oct. 19 2007.

[2] P. Kauff, K. Müller, A. Smolic, et al. "Depth map creation and imagebased rendering for advanced 3DTV services providing interoperability and scalability," Signal Proc.: Image Comm, no. 2, 2007.

[3] M. Domański, M. Gotfryd, K. Wegner, "View synthesis for multiview video transmission", The International Conference on Image Processing, Computer Vision, and Pattern Recognition, Las Vegas, USA, 13-16 July 2009.

[4] N. A. Manap, J. J. Soraghan, "Novel view synthesis based on depth map layers representation", 3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video 2011.

[5] "View synthesis algorithm in view synthesis reference software 3.0 (VSRS3.0)," ISO/IEC JTC1/SC29/WG11, Geneva, Switzerland, Tech. Rec. M16090, Feb. 2009.

[6] J. Luo, K. Qin, Y. Zhou, M. Mao, R.Li, "GPU-based multi-view rendering for spatial-multiplex autostereoscopic displays," Computer Science and Information Technology (ICCSIT) 9-11 July 2010.

[7] Y. Zhu, T. Zhen, "3D Multi-view Autostereoscopic Display and Its Key Technologie," Information Processing, 2009. APCIP 2009. Asia-Pacific Conference on , 18-19 July 2009.

[8] M. Bertalmio, A. L. Bertozzi, G. Sapiro, "Navier-Stokes, Fluid Dynamics, and Image and Video Inpainting", Proceedings of the International Conference on Computer Vision and Pattern Recognition ,Dec. 2001.

[9] Y. Morvan, P. H. N. de With, D. Farin , Platelet-based coding of depth maps for the transmission of multiview images In Proceedings of SPIE: Stereoscopic Displays and Applications, Vol. 6055 (2006).

[10] G. Tech, K. Wegner, Y. Chen, S. Yea, "3D-HEVC Test Model 3″ , Joint Collaborative Team on 3D Video Coding Extension Development of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11 Doc. JTC3V-C1005, 3rd Meeting: Geneva, CH, 17‑23 January 2013.

[11] O. Stankiewicz, K. Wegner, M. Domanski, "First version of depth maps for Poznan 3D/FTV test sequences", ISO/IEC JTC1/SC29/WG11, MPEG2010/M17176, Kyoto, Japan, January 2010.

[12] "Call for Proposals on 3D Video Coding Technology", ISO/IEC JTC1/SC29/WG11 MPEG2011/N12036, Geneva, Switzerland, March 2011.

[13] Y.-S. Ho, E.-K. Lee, C. Lee "Video Test Sequence and Camera Parameters" ISO/IEC MPEG M15419, Archamps, France, April 2008.

[14] D. Rusanovskyy, P. Aflaki, and M. M. Hannuksela, "Undo dancer 3DV sequence for purposes of 3DV standardization," ISO/IEC JTC1/SC29/WG11, Geneva, Switzerland, Tech. Rec. M20028, Mar. 2011.

[15] J. Zhang, R. Li, H. Li, D. Rusanovskyy, and M. M. Hannuksela, "Ghost town fly 3DV sequence for purposes of 3DV standardization," ISO/IEC JTC1/SC29/WG11, Geneva, Switzerland, Tech. Rec. M20027, Mar. 2011.

[16] M. Tanimoto, T. Fujii, M. P. Tehrani, M. Wildeboer, N. Fukushima, H. Furihata, "Moving Multiview Camera Test Sequences for MPEG-FTV",ISO/IEC JTC1/SC29/WG11 M16922, Xian, China, October 2009.