View Extrapolation Method using Depth Map for 3D Video Systems

Cheon Lee and Yo-Sung Ho
Gwangju Institute of Science and Technology (GIST)
261 Cheomdan-gwagiro, Buk-gu, Gwangju, 500-712, Republic of Korea
E-mail: {leecheon, hoyo}@gist.ac.kr, Tel: +82-62-715-2263

***Abstract*— The next generation 3D video systems employ the view synthesis method that generates virtual viewpoint images using depth data to provide realistic and comfortable depth impression. It is suitable for the advanced stereoscopic displays or the auto-stereoscopic displays since it can generate multi-view images. To provide high-quality virtual view images, the 3D video system should minimize the visual artifacts induced by the hole regions. Particularly, the visual artifacts arise severely during view extrapolation due to lack of information from the reference views. In this paper, we propose an efficient extra-view generation method, which is useful for the auto-stereoscopic displays in the circumstance that the number of provided viewpoint is not sufficient at the displays. First, we propose a framework that generates outer viewpoint images. Second, we propose an efficient hole filling methods in spatial and temporal directions. Using objective and subjective evaluations, we showed that the proposed method generate high-quality extra-viewpoint images.**

I. INTRODUCTION

New generation of three-dimensional video (3DV) is coming. In fact, research on 3D imaging has a long history from the studies on stereopsis. It is the process in human visual system leading to the perception of depth impression from the two slightly different projections of the world onto retinas of the two eyes; it was the first description by Charles Wheatstone in 1838 [1]. He invented a stereoscope to show that stereopsis can be realized by creating illusion of depth from flat pictures that includes differences in horizontal disparity. In recent years, research efforts have been strengthened due to the advanced of IT technologies from capturing to display.

There are two main issues on 3D video technologies, i.e., the data compression on multi-view video data including supplementary data and the 3D rendering without visual fatigues. Since the amount of 3D data is greater than a single viewpoint video, an efficient video coding for 3D video contents is highly required; it is a core technology in 3D video service. Recently, the moving picture experts group (MPEG), which is working group to set standards for audio and video compression and transmission, started activities on 3D video systems since 2001 [2].

As a first step of standardization, they explored various technologies related on 3D video in the name of 3D audio-visual (3DAV). After exploring related 3D technologies, the multi-view video coding (MVC) was developed which compresses multiple viewpoint video under the joint video team (JVT) in 2007; it was the first phase of FTV (free-viewpoint TV) work. Afterward, as a second phase of FTV work, the standardization activity on 3D video coding (3DVC) has started in 2008. The primary goal of 3DVC is to define a data format and associated compression technology to enable the high-quality reconstruction of synthesized views for various types of 3D displays [3].

For comfortable and safe 3D rendering, selecting optimal baseline is very important; it highly depends on the disparity range of a 3D scene and viewing position of a user. Assuming that a user can adjust the depth range in manual, generating virtual viewpoint images and selecting proper views can be a good approach to provide optimal 3D scene.

Generating additional view uses depth data for shifting viewpoint from the reference view to the virtual view. Regarding the position of the virtual view, we can classify the generation method into view interpolation and view extrapolation. For the intermediate viewpoint between the reference views, we use the view interpolation method, while the outer viewpoints use the *view extrapolation*. For example, based on Fig. 1, the reference views are V_1, V_4, and V_7; hence the intermediate views, V_2, V_3, V_5, and V_6 use the view interpolation and the outer views V_{-1}, V_0, V_8, and V_9 use view extrapolation.

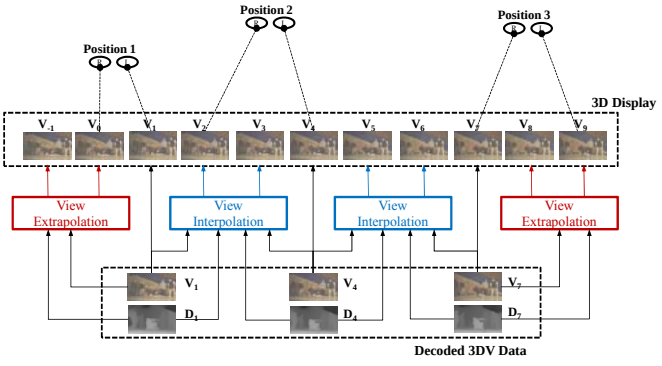

Fig. 1. 3D rendering using view interpolation and view extrapolation

Both view generation methods are based on the 3D image warping (these will be described in the following section), but uses different hole filling method. For the case of view

interpolation, hole filling is rather easy because most of the holes are can be found at the other reference view except for the common hole [4],[5]. However, for the case of view extrapolation, there is no referable information in the reference view. Therefore, an efficient hole filling method is strongly necessary to generate high-quality outer viewpoint image [6]. In this paper, we propose a framework for view extrapolation and an efficient hole filling methods for view extrapolation.

II. View Extrapolation Method

A. 3D Image Warping for Viewpoint Shifting

The 3D image warping is a basic technique that shifts the viewpoint using depth map. By defining the pixel correspondences between views using depth information, we obtain a synthesized image at the virtual viewpoint. Since the depth value describes the distance between the camera and objects in the scene, we can find the corresponding 3D point of a pixel in the reference image. Re-projecting the 3D point to the virtual viewpoint, we can find the geometrical relations between the references and virtual views.

Using this pixel correspondence between the reference view and the virtual view, we obtain the viewpoint shifted image. Mapping the correspondent colors from the reference view to the virtual view, we can obtain a synthesized image; it is called forward warping. However, in practice, we use backward warping as shown in Fig. 2; we warp the depth image to the virtual viewpoint and then we obtain the color image of the virtual view. This method reduces the effect of small holes. In detail, some pixel in the virtual view can be mapped two or more corresponding pixels from the reference view due to the rounding operation, while other pixels have no corresponding pixel. This one-pixel-with hole is called small hole, and it can be filled with a simple median filter. Since a depth image has higher spatial correlations than a color image, the backward warping is better than the forward warping.

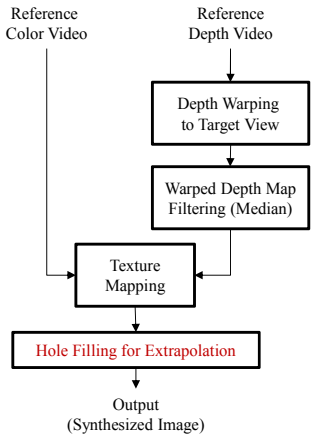

Fig. 2. Flowchart of view extrapolation

B. Hole Filling Method for View Extrapolation

The most important process is the hole filling on the viewpoint shifted color image. As we mentioned above, the hole area is a newly revealed area at the virtual viewpoint. If there are more than two reference views, the hole filling process is much easier because the corresponding area for the hole area might exist at the other reference view. However, view extrapolation can use only one reference view; hence it is much difficult to fill the holes without generating visual artifacts and temporal flickering. In order to solve this problem, we propose an efficient hole filling method for view extrapolation as shown in Fig. 3. We use three data, the synthesized color image, the synthesized depth map, and the alpha map; these three data are available after the texture mapping.

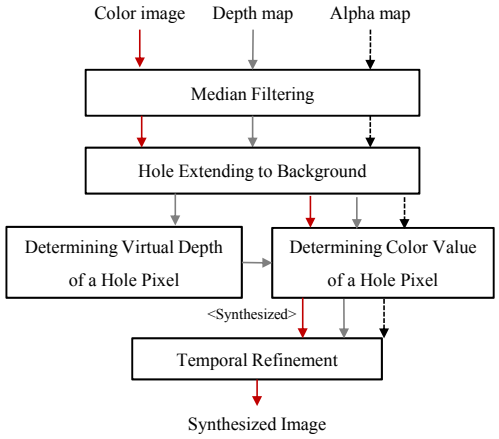

Fig. 3. Hole filling method for extrapolation

The first step is reducing the useless textures existing in the hole area. Some texture pixels are positioned at the middle of hole area due to the false depth value. Even though these can be used for the hole filling, its significance is rather small. Therefore, we remove them from the synthesized image using the median filter by conducting on the alpha map. We reflect the results to both synthesized color and depth images. Since the next process is hole extending to background, we need to distinguish the background boundary from the hole area. If the hole pixels exist in the middle of hole area, it is hard to determine the background area.

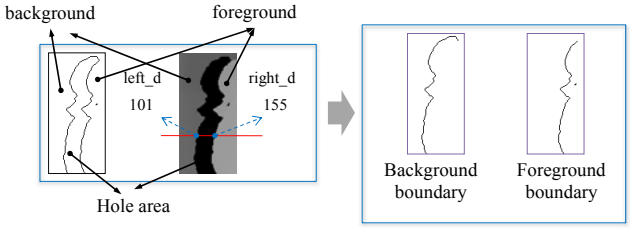

Fig. 4. Extraction of background boundary by comparing surrounded depth values

A depth map can be obtained in various approaches, e.g., using a TOF (time-of-flight) camera or a depth estimation

algorithm. Estimating the depth value of object boundary is very hard problem and it may induce the boundary noise problem; we have already proposed a solution for this [4]. To remove such noise with a simple operation, we extend the hole area toward background. We distinguish the background boundary from the hole boundary by comparing the depth values around hole area as shown in Fig. 4. If the left depth value is 101 and the right depth value is 155, respectively, the left one is located at the background; hence we determine that the left pixel is in the background boundary. In this manner, we obtain the background boundary as shown in the right image of Fig 4. Extending the hole area toward background is very simple. If the Euclidian distance of pixel is below than $\sqrt{2}$, we regards it as a hole.

The next process is determining color values of the hole area. Here are some assumptions for hole filling. First, the hole area would be the extended region of background. Second, therefore, the actual depth value of the hole area would have the similar value to the background. Third, the depth value of background exists around hole region. Based on these assumptions, we designed a bilateral filter determining a color value for a hole pixel. Following the third assumption, we choose a virtual depth value $\hat{d}$ for a target hole pixel as described in Eq. (1); the minimum depth value is regarded as a virtual depth representing background.

$$\begin{cases} \hat{d} = \min D(u,v) \\ D(u,v) \in W \end{cases} \tag{1}$$

where D and W denote the depth image and a block-based window, respectively.

To determine a proper color for the common hole, we consider three data as we mentioned above. In order to discard the colors of foreground objects, we use the depth information of neighboring reference pixels. The close objects are more important in the sense of color. Hence we designed an efficient hole filling filter as Eq. (2) which is modified from the typical bilateral filter. Let the synthesized image, the alpha map, and the depth map be I, α, and D, respectively, and bilateral filter radius be r. For a typical pixel p = {x, y}, assume $\vec{u}_p$ = {x − r, · · ·, x + r}, $\vec{v}_p$ = {y − r, · · ·, y + r}, we determine the color of the common hole C as:

$$C(x,y) = \frac{\sum_{u \in \vec{u}_p} \sum_{v \in \vec{v}_p} W(u,v,\hat{d}) \cdot C(u,v)}{\sum_{u \in \vec{u}_p} \sum_{v \in \vec{v}_p} W(u,v,\hat{d})} \tag{2}$$

$$W(u,v,\hat{d}) = \exp\left(\frac{\left\| \hat{d}, D(u,v) \right\|^2}{2\sigma_D^2} \right) \exp\left(\frac{(x-u)^2 + (y-v)^2}{2\sigma_r^2} \right) \tag{3}$$

where σ_D^2 and σ_r^2 denote the standard deviations for depth similarity and range smoothness.

The proposed bilateral filter considers only the available pixels referring to the alpha map as:

$$C(u,v) = \alpha(u,v) \cdot I(u,v) \tag{4}$$

The whole process performs iteratively until the whole hole area are filled completely. With one operation, pixels of background boundary are determined using the proposed filter, and then update three data. As a result, the hole area are filled iteratively from background to foreground.

Above proposed hole filling is designed for one still image; it does not consider temporal consistency. Temporally inconsistent virtual viewpoint view may have flickering artifacts; it is a several artifacts for 3D viewing. Therefore, we designed a temporal refinement method that consists of two methods: availability of the hole area and fluctuation of depth value in both previous and current images. First, we check the usability of the previous frame for a hole area using both alpha maps, $\alpha_{t-1}$and α_t. We copy the texture of hole area from the previous frame if there are valid textures which is visible (non-hole). Second, in the case that two frames have the same hole area between the previous and the current frame, we copy the texture of the previous frame if the depth value of current frame of the hole area is higher than that of the previous frame. This method reduces the temporal flickering

III. Experimental Results

We evaluated the proposed method using both subjective and objective measures for the sequences provided MPEG 3DV adHoc group [7]. Table 1 presents the test sequences and views for testing. The 3DV group has developed the view synthesis reference software (VSRS) for generating intermediate view images using two reference views [8]; which is one of popular approaches in depth-image-based-rendering (DIBR) [9]. Based on the structure of VSRS, we implemented the proposed extrapolation method to compare the quality of results In VSRS, the implemented hole filling is the inpainting method which determines the color for a hole using neighboring pixels [10].

Table 1. Testing conditions for experiments

Test Sequences	Reference Views (O_L-O_R)	Synthesized Views (S_{OL}-S_{OR})	Image Size	Total Frames
Book_arrival	10-8	11-7	1024x768	100
Newspaper	4-6	3-7	1024x768	300
Cafe	2-4	1-5	1920x1080	300

For subjective evaluation, we compared the visual artifacts of the synthesized images. From Fig. 5 to Fig. 7, we show the comparison of the experimental results. All left image is the original image for each view, the entire center image is the result of VSRS, and entire right image is the result of the proposed method. As a result, the results of VSRS show some visual artifacts around the object boundaries; these are because that the inpainting method considers only color values around the holes, while the proposed method considers the depth value and extend the colors of the background. The proposed method discards the colors of the foreground objects clearly, hence the hole area are filled with the similar color to

the background object. Since the proposed method uses the modified bilateral filter, the hole filled area show rather blurred texture. In addition, in Fig. 7, we can see the background texture of around his head is very clear; it is the improved result of the temporal refinement.

For objective evaluation, we compared the PSNR values for both methods. Since all synthesized view has its own original view captured by multiple cameras, we calculated PSNR values. Table 2 presents the comparisons of PSNR values. Some views were improved, but the other views were worsened. The average quality for four sequences has been increased as much as 0.72 dB. Although some views showed lower PSNR values comparing to the VSRS results, the improvement of subjective quality is obvious through the experiments. We expect that this method can be useful for achieving the advanced 3D video system for future broadcasting technology.

(a) original (b) VSRS3.5 (c) proposed

Fig. 5. Comparisons of synthesized images of 'Book_arrival' sequence

(a) original (b) VSRS3.5 (c) proposed

Fig. 6. Comparisons of synthesized images of 'Newspaper' sequence

(a) original (b) VSRS3.5 (c) proposed

Fig. 7. Comparisons of synthesized images of 'Cafe' sequence

Table 2. Results of objective evaluation using PSNR

Test Sequences	Synthesized View	PSNR (dB) VSRS3.5 (a)	PSNR (dB) Proposed (b)	Δ PSNR (dB) (b) – (a)
Book_arrival	11	35.50	35.83	+0.33
	7	34.54	34.33	-0.21
Newspaper	3	26.86	26.70	-0.16
	7	31.98	31.62	-0.36
Cafe	1	30.73	33.39	+2.66
	5	30.61	32.66	+2.06

IV. Conclusion

The multi-view image generation is an essential technology for providing comfortable and realistic 3D viewing. When we generate a virtual viewpoint image, hole regions arise due to the lack of information in the reference view. In this paper, we described the view extrapolation method using depth map and efficient hole filling methods. Since the view extrapolation method refers to only one reference view, we need to determine the alternative textures considering the neighboring pixels. We first determined the color values of the hole region using the proposed bilateral filters which considers the similarity of depth values for distinguishing distance of objects. Basically, we neglected the colors of foreground objects by comparing the depth values. Then, we enhanced the temporal consistency using the proposed temporal refinement method. By the experiments, the visual artifacts were significantly reduced for the hole areas. Since the proposed method generates high-quality virtual viewpoint images, we expect that it is useful for both the advanced stereoscopic displays and the auto-stereoscopic displays.

Acknowledgment

This research was supported in part by MKE under the ITRC support program supervised by NIPA (NIPA-2011-(C1090-1111-0003)).

References

[1] C. Wheatstone, "On Some Remarkable, and Hitherto Unobserved, Phenomena of Binocular Vision," *Philos. Trans. R. Soc.,* vol. 54, pp. 196-199, 1838.

[2] A. Smolic, K. Mueller, P. Merkle, C. Fehn, P. Kauff, P. Eisert, and T. Wiegand, "3D Video and Free Viewpoint Video - Technologies, Applications and MPEG Standards," in *IEEE ICME*, Canada, pp. 2161-2164, 2006.

[3] ISO/IEC JTC1/SC29/WG11, "Call for Proposals on 3D Video Coding Technology," *in MPEG output document* N12036, March 2011.

[4] C. Lee and Y. S. Ho, "View Synthesis using Depth Map for 3D Video," *APSIPA ASC 2009*, pp. 350-357, 2009.

[5] ISO/IEC JTC1/SC29/WG11, "Common-hole Filling for Boundary Noise Removal in VSRS," *in MPEG input document* M18514, Oct. 2010.

[6] ISO/IEC JTC1/SC29/WG11, "Report on Experimental Framework for 3D Video Coding," *in MPEG output document* N11631, Oct. 2010.

[7] ISO/IEC JTC1/SC29/WG11, "Description of Exploration Experiments in 3D Video Coding," *in MPEG output document* N11831, Jan. 2011.

[8] VSRS. *View Synthesis Reference Software*. Available: http://wg11.sc29.org/svn/repos/MPEG-4/test/trunk/3D/view_synthesis/VSRS

[9] C. Fehn, "Depth-Image-Based Rendering (DIBR), Compression and Transmission for a New Approach on 3D-TV," pp. 93-104, 2004.

[10] A. Telea, "An Image Inpainting Technique based on The Fast Marching Method," *Journal Graphics Tools,* vol. 9, pp. 25-36, May 2004.