The 2008 International Conference on Embedded Systems and Intelligent Technology
February 27-29, 2008, Grand Mercure Fortune Hotel, Bangkok, Thailand

MULTI-VIEW IMAGE GENERATION FROM DEPTH MAPS AND TEXTURE IMAGES USING ADAPTIVE SMOOTHING FILTER

Sang-Beom LEE, Sung-Yeol KIM and Yo-Sung HO

Gwangju Institute of Science and Technology (GIST)
E-mail: {sblee, sykim75, hoyo}@gist.ac.kr

Since depth-image-based rendering (DIBR) is one of the rendering processes to generate virtual views from the texture image and the depth map, weinvestigate DIBR as a core technique for three-dimensional television. In this paper, we propose a new scheme to generate multi-view images using DIBR. In order to improve visual quality of multi-view images at newly exposed areas, we preprocess the depth map using an adaptive smoothing filter. In conventional algorithms, the depth map is collapsed since they applied a filter to the whole depth map. After extracting object boundaries from the depth map, we apply smoothing filters with different window sizes according to the strength of depth discontinuities. With the proposed scheme, we can not only maintain reliable depth map qualities, but also generate high-quality multi-view color images. Experimental results show that our scheme outperforms previous works in multi-view image generation and supports various functionalities.

The 2008 International Conference on Embedded Systems and Intelligent Technology
February 27-29, 2008, Grand Mercure Fortune Hotel, Bangkok, Thailand

Multi-view Image Generation from Depth Map and Texture Image Using Adaptive Smoothing Filters

Sang-Beom Lee, Sung-Yeol Kim, and Yo-Sung Ho
Gwangju Institute of Science and Technology (GIST)
E-mail: {sblee, sykim75, hoyo}@gist.ac.kr

Abstract

Since depth-image-based rendering (DIBR) is one of the rendering processes to generate virtual views from the texture image and the depth map, we investigate DIBR as a core technique for three-dimensional television. In this paper, we propose a new scheme to generate multi-view images using DIBR. In order to improve visual quality of multi-view images at newly exposed areas, we preprocess the depth map using an adaptive smoothing filter. In conventional algorithms, the depth map is collapsed since they applied a filter to the whole depth map. After extracting object boundaries from the depth map, we apply smoothing filters with different window sizes according to the strength of depth discontinuities. With the proposed scheme, we can not only maintain reliable depth map qualities, but also generate high-quality multi-view color images. Experimental results show that our scheme outperforms previous works in multi-view image generation and supports various functionalities.

1. Introduction

Owing to significant advancements in computing power, interactive computer graphics, immersive displays, and digital transmission, we experience and reproduce simulations of reality. Human-computer interactions, computer-generated haptic data and kinesthetic interfaces provide us with experience of the virtual world. Interactive computer graphics also allow us direct and multi-sensory experiences. Especially, advances in display devices have been aimed at improving the range of vision, such as high-definition or immersive displays. They provide us with a feeling of 'being there', or presence, from the simulation of reality [1].

Three-dimensional television (3DTV) is one of the next-generation broadcasting systems that can provide us with the feeling of presence. Especially, multi-view image generation is in the spotlight as a core technology of 3DTV.

As shown in Fig. 1, there are various approaches to generate multi-view images, especially, multi-view camera system and depth image-based rendering (DIBR) system [2]. The multi-view camera system employs a finite number of cameras to obtain wide-viewing angle images. By displaying those multi-view images on auto-stereoscopic monitors, we can feel more reality. However, the multiple camera system requires much more complex encoding and transmission techniques than the single view case. Although the multi-view camera system provides us with wide-angle views, it is difficult to send the contents generated by a multi-view camera system to the receiver through a network of limited bandwidth.

On the other hand, the DIBR system only uses two types of images: the texture image and the depth map. With DIBR techniques, we can render arbitrary virtual views from scenes reconstructed by the texture image and depth map. Although DIBR systems have limitations of narrow-viewing angles, they are considered as a suitable main theme for 3DTV since we only need two streams to support all the functionalities of multi-view image generation. Furthermore, the depth map coding is more efficient than natural image coding due to low variances of the depth values. 3DTV systems using DIBR can not only enable us to reduce the bandwidth, but also synthesize new images as if they were captured at arbitrary view points. In other words, since the whole depth values in the original viewpoint are known, a synthesized image at a virtual viewpoint can be rendered by 3D warping [3].

Although multi-view image generation is essential for 3DTV, there are some problems. One of the most significant problems is that there is no information at newly exposed areas, which are occluded in the original view but become visible in any of the virtual views, called as disocclusion. Therefore, we should remove those areas efficiently so that the synthesized images look more natural. Another problem is related to variations of the depth information, called as depth discontinuity. Depth discontinuity also makes us hard to generate synthesized images.

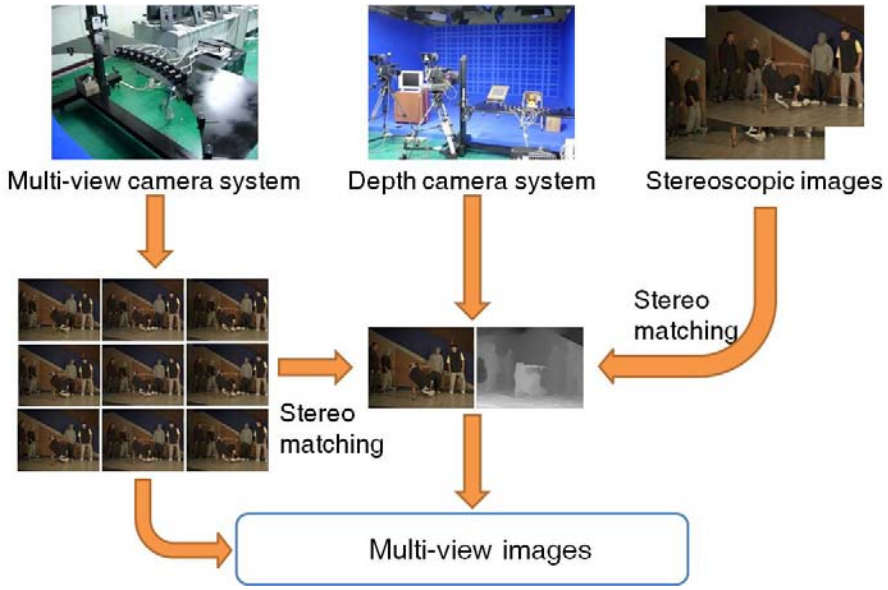


Figure 1. Various approaches to obtain multi-view images

In this paper, we try to solve problems in DIBR and propose a new multi-view image generation system with minimized degradation of depth maps. In order to reduce both disocclusion areas and geometric distortions, we apply a Gaussian filter to the depth map only near the boundaries of objects. In addition, we apply a Sobel filter in advance to detect the boundaries of objects. When we find the boundaries, the simple depth map is used as an input of the Sobel filter. Moreover, the filtered areas are altered by the strength of depth discontinuity. All pixels in the texture image and the depth map are used as the geometric and photometric components of the mesh modeling through a mesh triangulation for precise rendering.

2. Previous Multi-view Image Generation

2.1. Previous Hole-filling Techniques

When occlusion areas are occurred while synthesizing the virtual views, we can easily resolve that problem: the points that are farther away from the virtual viewpoint are substituted by the nearer points. However, when disocclusion areas are detected, the problem becomes more complex since there is no information about the disoccluded area.

In DIBR, several techniques to deal with the holes have been introduced. Those methods are categorized into two approaches: filling out holes by using nearby useful color information, or preprocessing of depth map. Typical techniques for the hole-filling include linear interpolation of foreground and background image color, background color extrapolation, mirroring of background color information, and smoothing of depth information with a Gaussian filter.

One of the solutions to resolve disocclusion artifacts is the layered depth image (LDI) [4]. LDI stores more than one pair of associated color and depth values for each pixel, with the number of layers typically depending on the scene complexity as well as the required synthesis quality. Although LDI seems to be a proper substitute, it has a main disadvantage that it is complicated too much for all parts, such as coding, transmission, and synthesis.

Another solution is the mesh-based scene representation using OpenGL. The disocclusion areas are filled automatically by linear surface interpolation. However, this approach causes rubber sheet artifacts at the virtual viewpoint.

2.2. Asymmetric Smoothing of Depth Map

We notice that the virtual view after smoothing of depth maps still has geometric distortions even if the image quality of the virtual view is quite better than that of the virtual view without preprocessing. From the result, analysis of underlying reasons for the geometric distortions suggests that the strength of smoothing the depth map in the horizontal direction should be less than that of smoothing in the vertical direction. Therefore, vertical objects have similar depth values throughout after depth smoothing. We call this asymmetric smoothing.

Recently, a new method for depth map pre-processing has been proposed [5] [6]. The flowchart of the algorithm is illustrated in Fig. 2. The basic approach of the algorithm is smoothing depth maps by using the asymmetric Gaussian filter. First, the depth map is smoothed using the filter. Then, 3D image warping is used to change the viewpoint and the hole-filling algorithm is applied in the last step.

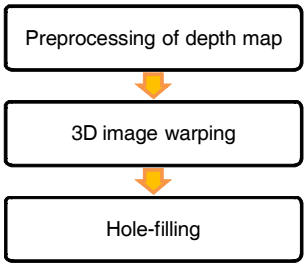


Figure 2. Flowchart of the preprocessing

2.3. Hierarchical Natural-textured Mesh

A new scheme to represent 3D natural videos has been proposed, called as a hierarchical natural-textured mesh stream (HNTMS) [7], using a DIBR technique. This scheme has a functionality to render 3D nature videos progressively. In addition, HNTMS enables us to deal with mesh-based depth representation more easily, when we perform 3D signal processing with its data while maintaining high-speed rendering.

Basically, HNTMS exploits the depth map and the texture image as the input data. The depth map can be obtained by a depth camera [8] directly or stereo matching techniques [9] [10]. In HNTMS, the depth map is divided into four layers: the number of layers (NOL), a grid, feature points, and object boundary layers. Then, 3D natural video is generated by a base layer and several enhancement layers. Finally, data in HNTMS are coded separately and transmitted to a receiver side according to users' capabilities and network conditions. Figure 3 shows an example of HNTMS with a hierarchical structure.

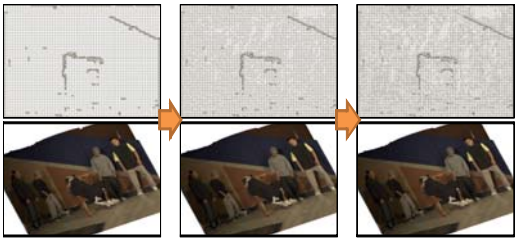


Figure 3. HNTMS with hierarchical structure

3. Proposed Preprocessing of Depth Map

3.1. Smoothing nearby the Boundaries

The most important issue of DIBR is how to deal with disocclusion areas. Mesh-based depth representation such as HNTMS uses a linear color interpolation to fill out holes. However, this method causes geometric distortions, called as rubber-sheet artifacts stretching from the boundary of the foreground to the background.

When the depth map is smoothed out, various multimedia applications which employ the depth information cannot use it efficiently. In other words, the deformation of depth qualities from a depth map preprocessing should not be ignored, even if the decoded depth map at the receiver side has already included some distortions, such as blocking artifacts, optical noises, and shape distortions. Therefore, deformation of the depth maps should be minimized while the depth map is preprocessed.

Our proposed scheme for depth map pre-processing is mainly composed of two parts: applying a Sobel filter to a depth map to detect object boundaries and applying a Gaussian filter near the boundaries. Figure 4 shows the block diagram of the proposed scheme.

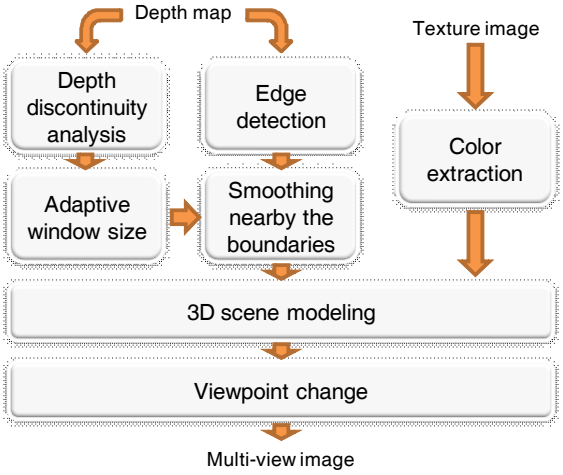


Figure 4. Block diagram

Most holes are usually generated where there are sharp depth discontinuities, that is, at boundaries of the objects. Aiming at this assumption, we reduce the filtered area. In the proposed preprocessing, boundaries of the objects are extracted from the depth map. In general, it is easy to use the depth map to extract the boundaries because the depth map is simpler than the color image. Then, the Sobel filter is applied to obtain the object boundaries. After finding the boundaries, the depth map is smoothed out only near the boundaries. The smoothing process is illustrated in Fig. 5.

Figure 5. Smoothing process

A 2D Gaussian smoothing filter for smoothing the depth map is defined by

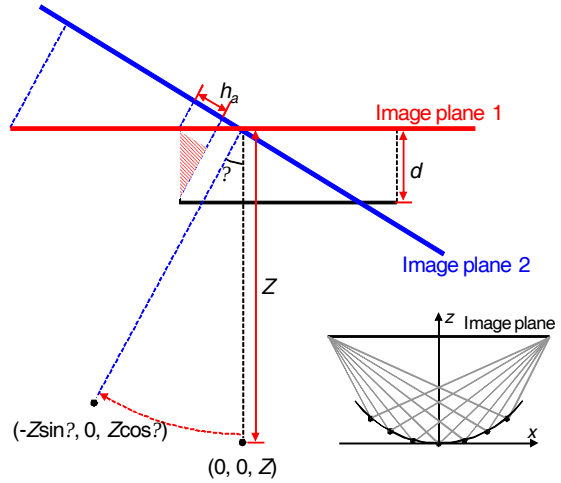
$$g(x,y) = \exp\left\{-\frac{x^2+y^2}{2\sigma^2}\right\} \quad (1)$$

$$G(x,y) = g(x,y) / \sum_x \sum_y g \quad (2)$$

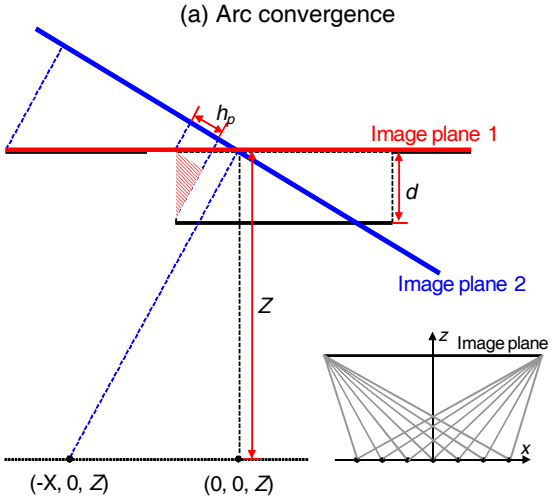
where x and y are in the range of $-w_x \le x \le w_x$, $-w_y \le y \le w_y$. In addition, w_x and w_y are window sizes in the horizontal and vertical direction, respectively. By our preprocessing, filtered areas can be minimized. As a result, we can prevent serious deformations of the depth map.

3.2. Gaussian Filter with Adaptive Window

Filtered areas can be reduced when we utilize different window sizes of the smoothing filter according to the strength of depth discontinuities. In other words, the smaller the filtered range is, the less deformation of the depth map is. Therefore, we need to derive the relationship between the length of holes and the strength of depth discontinuities so as to select a reasonable window size. Figure 6 shows the relationship between holes and depth discontinuities.



(a) Arc convergence



(b) Parallel convergence

Figure 6. Adaptive window size estimation

As shown in Fig. 6(a), we first derive the relationship when a virtual camera setup is the arc convergence, where θ is the rotation angle of the virtual camera. From the region of the triangle, we can easily derive

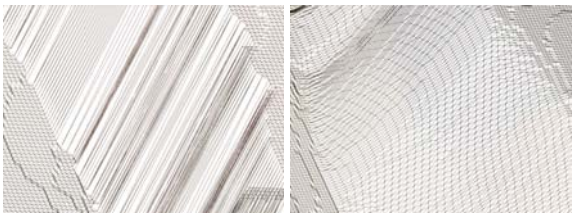
$$h_a = d \sin\theta \qquad (3)$$

Second, we derive the relationship when a virtual camera setup is parallel convergence, as shown in Fig. 6(b), where X is the camera distance and Z is the distance between the camera baseline and the original image plane. Similarly, we can derive

$$h_p = d \frac{X}{\sqrt{X^2 + Z^2}} \qquad (4)$$

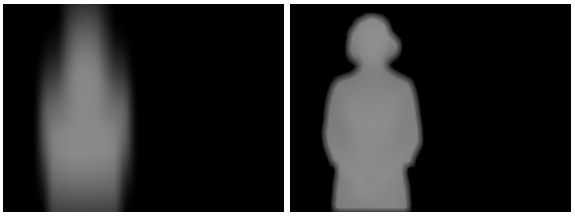
From Eq. 3 and Eq. 4, we notice that once we know depth discontinuity and movement of the viewpoint, we can easily estimate the length of holes. Finally, the window size w_x is equal to h.

Figure 7 shows the enlarged 3D models, and Figure 8 shows depth maps after preprocessing with an asymmetric filer and our adaptive smoothing filter. We notice that the boundary of the 3D model is smoothed, compared to the model that has no preprocessing. Moreover, the depth map filtered by an asymmetric filter is collapsed seriously, as shown in Fig. 8(a), while the depth map filtered near the object boundary is mostly preserved.



(a) No preprocessing (b) Proposed scheme

Figure 7. Comparison of 3D models



(a) Applying asymmetric filter (b) Proposed scheme

Figure 8. Comparison of depth maps

Since different depth maps contain different depth discontinuities, we cannot determine the constant viewing angle. Therefore, the viewing angles should be determined according to the maximum depth discontinuities in depth maps and this problem will be treated in future works.

4. Experimental Results and Analysis

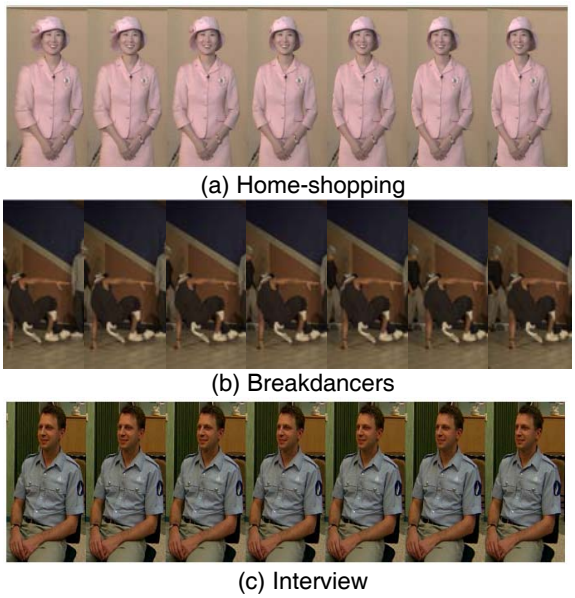
We have tested performances of the proposed depth map preprocessing using three pairs of depth maps and texture image sequences. The test sequences were “Home-shopping” with 720x486

resolutions, "Breakdancers" with 1024 x768 resolutions and "Interview" with 720x486 resolutions. "Home-shopping" sequences were captured from a depth camera, *Zcam*^{TM}, to make use of broadcasting contents for 3DTV in Realistic Broadcasting Research Center (RBRC) at GIST in Korea. "Breakdancers" sequence was provided by Microsoft as a test sequence for multi-view video coding (MVC) in MPEG. "Interview" was provided as a test sequence for the DIBR technique [11]. Among them, only depth maps of "Breakdancers" are obtained by the stereo matching algorithm.

We exploited depth maps and texture images to reconstruct 3D scenes. In this experiment, we used the whole texture and depth pixels to obtain more precise results. In a depth map, we can regard pixel positions and corresponding depth values as geometry information in the 3D domain. Pixel positions in the depth map were x and y coordinates and depth values were z coordinates. The photometry component could be obtained from the actual color of each pixel in the texture image.

4.1. Multi-view Image Generation

When we generated the multi-view images using the proposed scheme, a virtual camera was rotated around reconstructed 3D scenes from -15 to +15 degrees from the center viewpoint, where seven views were generated. Figure 9 shows the experimental results of multi-view image generation. We could reduce rubber-sheet artifacts since we applied the Gaussian filter to boundaries of the objects adaptively according to depth discontinuities. As a result, our system improved both overall image qualities of new views and depth qualities in comparison with previous algorithms.



(a) Home-shopping

(b) Breakdancers

(c) Interview

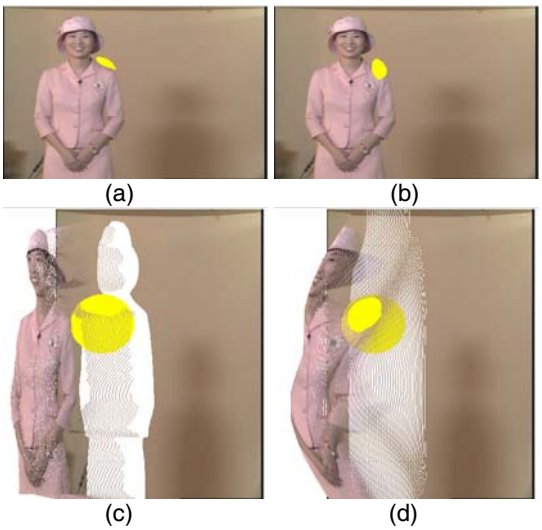
Figure 9. Multi-view generation results

Since X for parallel convergence and θ for arc convergence are altered by the viewpoint, holes h are also altered at different viewpoints. Ideally, the adaptive window size modeling can be performed for all viewpoints separately. However, if we consider the computational load, the modeling is performed for the viewpoint with the maximum distance and then estimated window sizes are applied to other viewpoints. In our experiment, the window size estimation was performed for the viewpoint with the maximum distance.

4.2. Depth Keying

We evaluated the performance of our scheme for depth keying. We chose the variation of a Gaussian smoothing filter to be equal to 10. The filter size was set to three times the variation of the filter. The filtered range was fixed for simplicity.

Figure 10 shows the result of simple depth keying. The sphere is set to be behind the woman model. Comparing Fig. 10(a) with Fig. 10(b), we noticed that the asymmetric filtering had a poor performance for depth keying. The sphere behind the woman was appeared in her body, as shown in Fig. 10(b) and Fig. 10(d), but this result should not be feasible in depth keying. On the other hand, the proposed scheme hid the sphere successfully and there is no irruption in the woman's body. As a result, our scheme was good for depth keying. In this experiment, the filtered range was fixed for simplicity.



(a) (b)

(c) (d)

Figure 10. Depth keying results

4.3. Performance Analysis

In order to evaluate the visual quality of the proposed scheme, we have made the original image at a virtual viewpoint using a computer graphic (CG) model. The comparison method is as follows. First, a texture image and its corresponding depth map are

extracted by a CG model. Second, using the CG model, the original color image at the virtual viewpoint is captured. Third, the color image and the depth map are rendered with mesh-based scene representation. Then, the synthesized image at the virtual viewpoint is captured. Notice that the virtual viewpoint is exactly the same as the previous one where the original image is captured. Finally, we compare these two images in terms of image quality. The comparison method is depicted in Fig. 11.

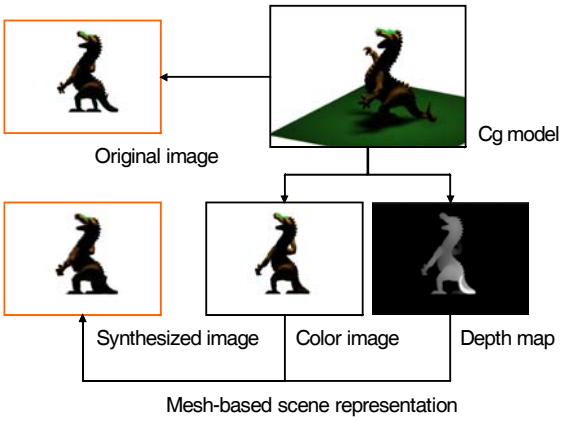


Figure 11. Performance comparison

The virtual camera is rotated by 5 degree, 10 degree, and 15 degree left from the center viewpoint. For simplicity, the filter size in the proposed scheme is fixed to be 10. Peak-Signal-to-Noise-Ratio (PSNR) is used as a comparison measure. The PSNR results are given in Table 1. We notice that the proposed scheme increased visual quality in comparison to the previous preprocessing method.

Table 1. PSNR results

Viewpoint	PSNR (dB)	
	Smoothing asymmetric filter	Proposed scheme
Left 5°	30.8417	32.9546
Left 10°	28.9481	31.8405
Left 15°	28.1003	30.9747

5. Conclusions

In this paper, we proposed a depth-image-based rendering (DIBR) technique to generate multi-view images. Since both the whole texture and depth information are used in generating the 3D mesh model, we can obtain a more precise representation of the natural 3D surface. We could resolve the disocclusion problem of DIBR efficiently using a smoothing filter with an adaptive window. Moreover, we could increase visual quality about 2.6 dB when we compared our scheme to the previous algorithm. As a result, we could not only generate multi-view images efficiently, but also use the depth map in 3D applications, since the depth information of object was preserved without serious loss.

Acknowledgements

This work was supported in part by ITRC through RBRC at GIST, and in part by MOE through the BK21 project.

References

[1] G. Riva, F. Davide, W.A. Ijsselsteijn, *Being There: Concepts, Effects and Measurement of User Presence in Synthetic Environments*, Ios Press, Amsterdam, Netherlands, 2003.

[2] C. Fehn, “Depth-Image-Based Rendering (DIBR), Compression and Transmission for a New Approach on 3-D TV,” *in Proceedings of SPIE Conf. Stereoscopic Displays and Virtual Reality Systems*, vol. 5291, pp. 93-104, Jan. 2004.

[3] William R. Mark, Leonard McMillan, Gary Bishop, “Post-Rendering 3D Warping", *in Proc. of Symposium on Interactive 3D Graphics*, pp. 7-16, April 1997.

[4] J. Shade, S. Gortler, L. He, and R. Szeliski, “Layered Depth Image," *in Proc. of SIGGRAPH'98*, pp. 231-242, July 1998.

[5] L. Zhang, W.J. Tam, “Stereoscopic Image Generation Based on Depth Images for 3D TV,” *IEEE Trans. on Broadcasting*, vol. 51, pp. 191-199, June 2005.

[6] W. J. Tam, A. Soung Lee, J. Ferreira, S. Tariq, and F. Speranza, “Stereoscopic Image Rendering Based on Depth Maps Created from Blur and Edge Information," *in Proc. of the SPIE: Stereoscopic Displays and Virtual Reality Systems*, vol. 5664, pp. 104-115, Jan. 2005.

[7] S.Y. Kim, S.B. Lee, Y.S. Ho, “Three-dimensional Natural Video System based on Layered Representation of Depth Maps,” *IEEE Trans. on Consumer Electronics*, vol. 52, pp. 1035-1042, Aug. 2006.

[8] 3DV Systems, http://www.3dvsystems.com/, 2005.

[9] D. Sharstein and R. Szeliski, “A Taxonomy and Evaluation of Dense Two-frame Stereo Correspondence Algorithms,” *in Proc. of IEEE Workshop on Stereo and Multi-Baseline Vision*, pp. 131-140, Dec. 2001.

[10] C.L. Zitnick, S.B. Kang, M. Uyttendaele, S. Winder, R. Szeliski, “High-quality Video View Interpolation Using a Layered Representation,” *in Proc. of SIGGRAPH’04*, pp. 600-608, Aug. 2004.

[11] A. Redert, M. Op de Beeck, C. Fehn, W. IJsselsteijn, M. Pollefeys, L. Van Gool, E. Ofek, I. Sexton, and P. Surman, “ATTEST: Advanced Three-dimensional Television System Techniques," *in Proc. of 3DPVT*, pp. 313-319, June 2002.