

## Boundary Filtering on Synthesized Views of 3D Video

Cheon Lee and Yo-Sung Ho

Gwangju Institute of Science and Technology (GIST)  
1 Oryong-dong, Buk-gu, Gwangju, 500-712, Republic of Korea  
E-mail: {leecheon, hoyo}@gist.ac.kr

### Abstract

When we generate an intermediate viewpoint image using multi-view images and their depth maps, we may have very annoying boundary noises in the background due to depth value errors around depth discontinuities. In this paper, we propose a boundary filtering method for synthesized images. After defining the boundary noise area using depth map, we replace them with corresponding texture information from the other reference image. Our experiments show that the proposed method improves visual quality of the synthesized virtual viewpoint images.

### 1. Introduction

3D video (3DV) is an emerging medium that can provide users with realistic contents based on multi-view video. It can be utilized in auto-stereoscopic display or free viewpoint TV (FTV). Users can enjoy natural and realistic 3D contents through those display devices. Therefore, generation technology of 3D contents using multi-view video is the key part of 3DV system. Image-based rendering (IBR) and depth-image based rendering are the relating research area.

Although many researches on generation of 3D content have been achieved, there was no killing application to service them until MPEG (moving picture experts group) working group started to investigate the 3D video system in December 2001. Standardization activity on multi-view video coding (MVC) has been worked since 2004, and WD (working draft) and its reference software has been distributed in April 2008 [1].

Differently from MVC, MPEG started the standardization of 3DV system since 2007. It basically employs video-plus-depth representation, thus it supports free viewpoint change and auto-stereoscopic display [2]. Test sequences were collected to setup the experimental environment in April 2008 [3], and those contents were evaluated by viewing test using stereoscopic display devices in July 2008 [4].

One important technology of 3DV system is generation of an intermediate viewpoint image. Among DIBR techniques, 3D warping is proper for view synthesis on 3DV system. An arbitrary viewpoint image can be rendered from any nearby viewpoint by two different forward and backward projection processes [5]. The backward projection needs the depth value of a pixel, thus the accuracy of depth determines quality of synthesized image. Unfortunately, depth information cannot be perfect because of ambiguity of depth discontinuity.

In this paper, we focus on inaccuracy of depth values around depth discontinuity. The unstable depth value causes the boundary noise near to the object boundary. It creates annoying defects around the boundary in the target viewpoint image. Therefore, we propose a boundary filtering method which extracts the boundary noises by replacing them with corresponding texture information referred to the other warped image.

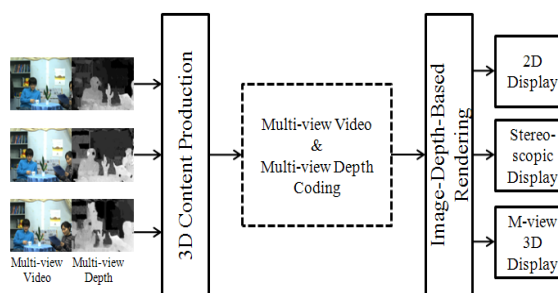


Fig. 1. 3D Video System

### 2. 3D Video System

In this chapter, we briefly introduce 3DV system and why we need the view synthesis method for the system.

As shown in Fig. 1, inputs of 3DV system are multi-view video and multi-view depth information. If the number of viewpoints of input video is  $M$ , the number of depth video is also  $M$ . In other words, every viewpoint image has its own depth video. Apparently, the encoder deals with two kinds of data: multi-view video and multi-view depth video.

The main functions of 3DV system are free viewpoints selection and auto-stereoscopic display. Accordingly, the view synthesis technique is a key technology which exploits the depth value to find pixel correspondence between the reference image and target image. The 3D warping is the most convenient method to find the correspondence. In the following chapter, we describe the whole process of view synthesis using 3D warping with its problems and solutions.

### 3. View Synthesis using 3D Warping

In this chapter, we briefly explain about the general view synthesis process using 3D warping technique.

#### 3.1. 3D Warping

View synthesis method under consideration is pixel-based texture mapping. If we can find the pixel correspondence between the target and reference images, we can reconstruct a scene at the desired viewpoint. Since 3DV system provides geometric information between views such as camera parameters and depth information, it is possible to define the pixel correspondence.

Camera parameter describes relationship between the camera coordinates and world coordinates. It consists of one intrinsic parameter  $\mathbf{A}$  and two extrinsic parameters: rotation matrix  $\mathbf{R}$  and translation vector  $\mathbf{t}$ . If a point  $\mathbf{X}$  in the real world projected to the pixel  $\mathbf{x}$  of an image, we can describe it using projective matrix  $\mathbf{P}=\mathbf{A}[\mathbf{R}|\mathbf{t}]$ .

When we try to find the correspondence of positions between the reference and target viewpoint images, we put the pixel position back into the world coordinates using Eq. (1).

$$\mathbf{X}_r = \mathbf{R}_r^{-1} \cdot \mathbf{A}_r^{-1} \cdot \mathbf{x}_r \cdot \mathbf{d}_r(\mathbf{x}_r) - \mathbf{R}_r^{-1} \cdot \mathbf{t}_r \quad (1)$$

where  $\mathbf{X}_r$  represents the position in the real world coordinates for a pixel  $\mathbf{x}_r$  in the reference image. The depth map of the reference viewpoint is  $\mathbf{d}_r$ . After this backward projection, we re-project it into the target camera using Eq. (2).

$$\mathbf{x}_t = \mathbf{P}_t \mathbf{X}_r \quad (2)$$

where  $\mathbf{x}_t$  is the corresponding pixel of  $\mathbf{X}_r$  in the target image projected by using the projective matrix  $\mathbf{P}_t$ . Repeating the backward and forward projection, we can map all pixels in the reference image into the target image. The synthesized image is obtained by copying color data from the reference image to the target view.



(a) From Left View (b) from Right View  
Fig. 2. Warped Images with Holes

#### 3.2. Texture Mapping

Using the 3D warping, we can define all pixel correspondences between the reference and target viewpoint images. The texture mapping process is coloring the target viewpoint image by following the pixels correspondence. The result images are described in Fig. 2. The white areas are newly exposed regions so called disocclusions. Since 3DV system allows using any viewpoint data, we can exploit any view. Since we use two neighboring reference images, we obtain two warped images by referring to the separate 3D warping processes.

#### 3.3. Hole filling

In this subsection, we explain the general procedure of the hole filling method in multi-view video.

As already explained above, hole area is a newly exposed region cause by 3D warping. When we change the viewpoint, some areas are disappeared and some are newly exposed. The reference image has no information about the appeared area. If we have only one viewpoint image and its depth map, it is hard to estimate. However, since 3DV system allows using the multi-view video and multi-view depth map, we can easily find the corresponding information of the hole area. The alternate texture exists at the other reference view image. Copying those information is our strategy to fill in the hole area.

### 4. Boundary Noise Filtering

#### 4.1. Boundary Noise

View synthesis using 3D warping is highly dependent on accuracy of depth map. Hence, unstable depth discontinuity around boundary creates boundary fraction noise. As shown in Fig. 3, fractions of the thumb exist in the background. It is because of the inaccurate depth value around the object boundary region. If the depth value for the depth discontinuity is perfect, the noise would not appear. However, the depth errors caused by the passive depth estimation are

inevitable because of ambiguity of estimation around depth discontinuity. When a user sees noises though the display, one may feel uncomfortable. Extracting it from the synthesized image is necessary to display more natural image.

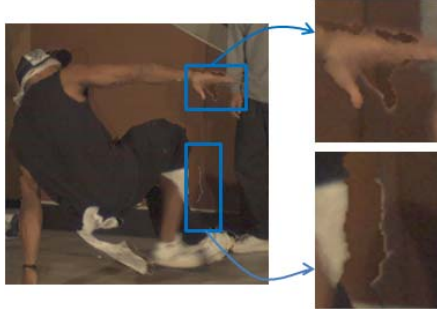


Fig. 3. Boundary Noise

The noise area is located near to the boundary of objects. After detecting the hole area, we choose the boundaries of background region. Then, we find the alternative texture information from the other reference view. Finally, we replace the noise area with the corresponding texture data. The whole process of boundary filtering is described in the Fig. 7.

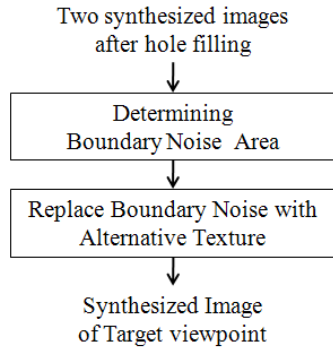


Fig. 4. Process of Boundary Noise Filtering

## 4.2. Extraction of Background Boundary

In order to extract the background boundary noise, we use the hole area as illustrated in Fig. 5. We find the boundary contour along the hole area. The top left image of Fig. 5 is the synthesized image referring to the left reference image. The top right image of Fig. 5 is the extracted boundary contour. By using the contour, we can divide the synthesized image into three regions: foreground, hole region and background. Since the hole region was filled in with referring to the other synthesized image, we don't touch anymore, but the boundary area of the background have to be filtered out because there is boundary noises.

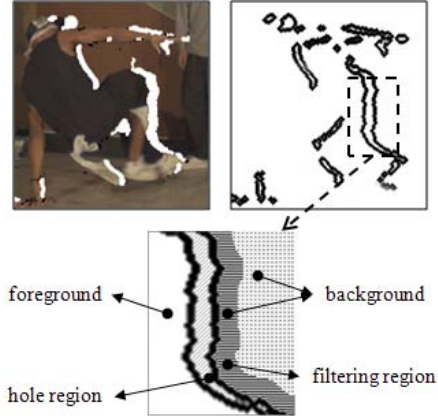


Fig. 5. Boundary Contour and Regions

## 4.3. Boundary Filtering

The issue region is adjacent region of background. In order to filter those background noise regions, we need to determine the boundary border. Since the depth values around the contour are able to compare, we define the contour having lower depth area as background contour. It is shown in the top left image in Fig. 6.

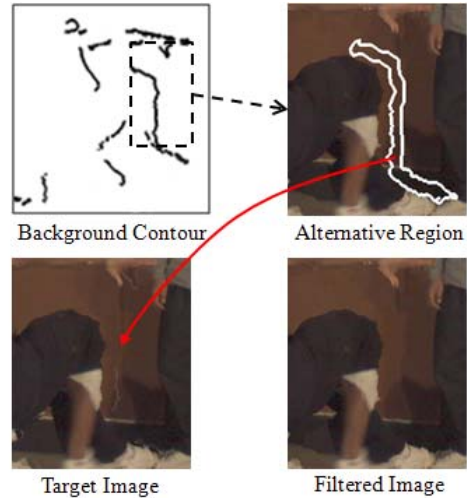


Fig. 6. Boundary Filtering

Next, we find the corresponding information at the other reference image. Since we are using multi-view video, the alternative texture information exists in the other reference view. We can find the texture data along the background contour as alternative information. By guiding the background contour, we copy the alternative information from the other synthesized image. Replacing the region of the target image with the alternative data generates the final synthesized image as shown in Fig. 6.

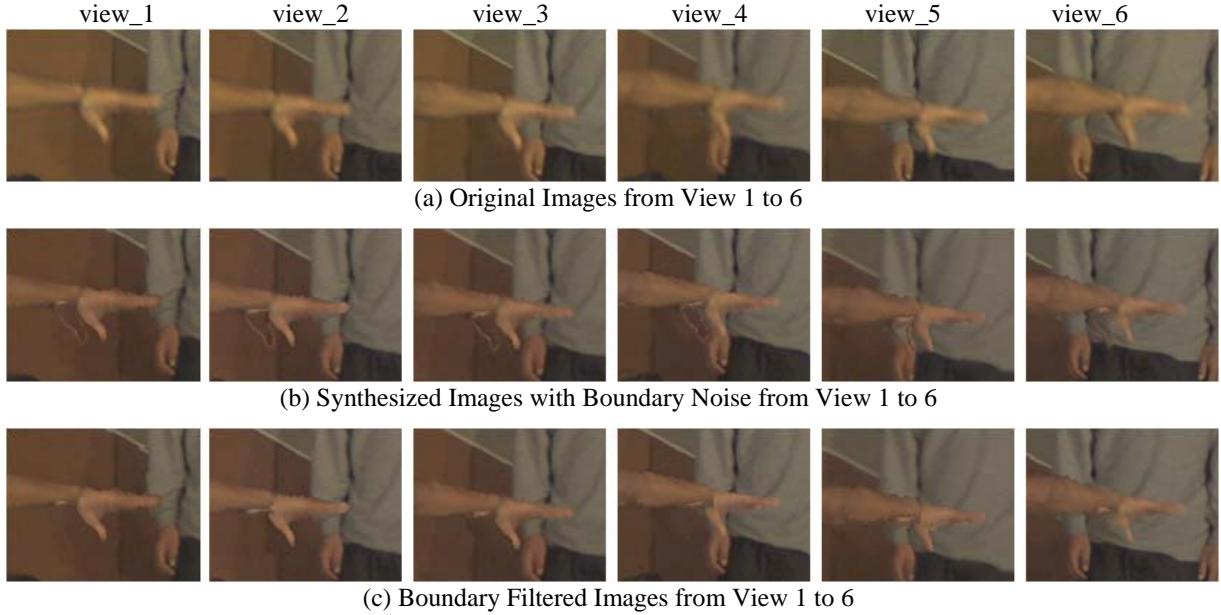


Fig. 7. Comparison of Synthesized Images

## 5. Experimental Results

We have experimented on ‘breakdancers’ sequences which is provided by Microsoft Research. They provided multi-view video and its depth maps. The depth map is generated by using segment based depth estimation. The details are described in Zitnick’s paper [6]. The total number of views is 8 and its file size is VGA (1024x768). We selected two reference views and synthesized the intermediate view image. Total number of synthesized views is 6 and the result images are demonstrated in Fig. 7. The top row images are the original images and the middle row shows the synthesized images with boundary noise. The bottom row images are the boundary filtered images.

Images in Fig. 7 (b) are the examples that are containing the boundary noise. You can see the shape of thumb at the background region. This region of boundary noise is commonly visible area. Our algorithm is to process that area having noise in the background. As shown in Fig. 7 (c), the noise is clearly filtered out.

## 6. Conclusion

We have described boundary filtering method on view synthesis for 3DV. The boundary noise is caused by the erroneous value of depth map around depth discontinuity. Assuming that the depth map is untouchable, we detect the boundary of objects and

extract the background boundary contour where exists noises. Since 3DV system allows using any viewpoint image, we can find the alternative texture for the boundary noise area. We find it from the other reference view and replace the boundary noise area with the corresponding texture information referring to the other viewpoint image. By experiments, we obtained synthesized images having clear background image.

## Acknowledgements

This work was supported in part by ITRC through RBRC at GIST (IITA-2008-C1090-0801-0017).

## References

- [1] JVT of ISO/IEC MPEG & ITU-T VCEG, JVT-AA212, “WD 1 Reference software for MVC,” April 2008.
- [2] ISO/IEC JTC1/SC29/WG11, “Introduction to 3D Video,” W9784, May 2008.
- [3] ISO/IEC JTC1/SC29/WG11, “Description of Exploration Experiments in 3D Video Coding,” W9783, May 2008.
- [4] ISO/IEC JTC1/SC29/WG11, “Results of 3D Video Expert Viewing,” W9992, July 2008.
- [5] L. McMillan, “An Image-based Approach to Three-dimensional Computer Graphics,” Technical report, Ph.D. Dissertation, UNC Computer Science TR97-013, 1999.
- [6] L. Zitnick, S.B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, “High-quality video view interpolation using a layered representation,” SIGGRAPH 2004, Aug. 2004.