INTERNATIONAL ORGANISATION FOR STANDARDISATION
ORGANISATION INTERNATIONALE DE NORMALISATION
ISO/IEC JTC1/SC29/WG11
CODING OF MOVING PICTURES AND AUDIO

ISO/IEC JTC1/SC29/WG11
MPEG2008/M15851
October 2008, Busan, Korea

Source: GIST (Gwangju Institute of Science and Technology)
Status: Proposal
Title: View Synthesis Tools for 3D Video
Author: Cheon Lee and Yo-Sung Ho

1. Introduction

This document presents a view synthesis tools which is developed in response to "Call for Contributions on FTV Test Material" [1] since the 82th Shenzhen meeting. Our tool employs pixel-by-pixel 3D warping, hole filling method, and boundary noise processing [2]. In the next chapter, we explain about the whole structure of view synthesis briefly. Since view synthesis SW from Nagoya Univ. is chosen as a reference SW, we compared experimental results.

2. View Synthesis for 3DV

View synthesis in 3DV system is a key technology which generates an intermediate viewpoint image using multi-view video and its depth map. A depth map allows reconstructing a scene into the real world because depth information indicates the distance between object and camera. In our view synthesis tool, we used 3D warping technique to map a pixel in the reference viewpoint into the target viewpoint because it is the technique that employs depth data in IBR(image-based rendering). The warped image has hole area which is the region having no information in the reference view image. We defined the hole into two kind of classes: small hole and large hole. The small hole can be filled in by referring to the neighboring pixel values. However, the large hole is newly exposed area, thus it should be filled in by referring to the other reference view image. Most of view synthesis methods go to blending procedure after hole filling, but we added one more process that is boundary noise processing. We eliminate the boundary noises around depth discontinuity using boundary noise processing, then we mix two intensities from two different synthesized images for one target pixel. Figure 1 shows the hole procedure of our view synthesis tool, and followings are the involved methods.

- Pixel-by-Pixel mapping using 3D warping
- Generation of depth map for target viewpoint
- Median filtering for small hole filling
- Wide hole filling by copying from the other synthesized image
- Boundary noise processing

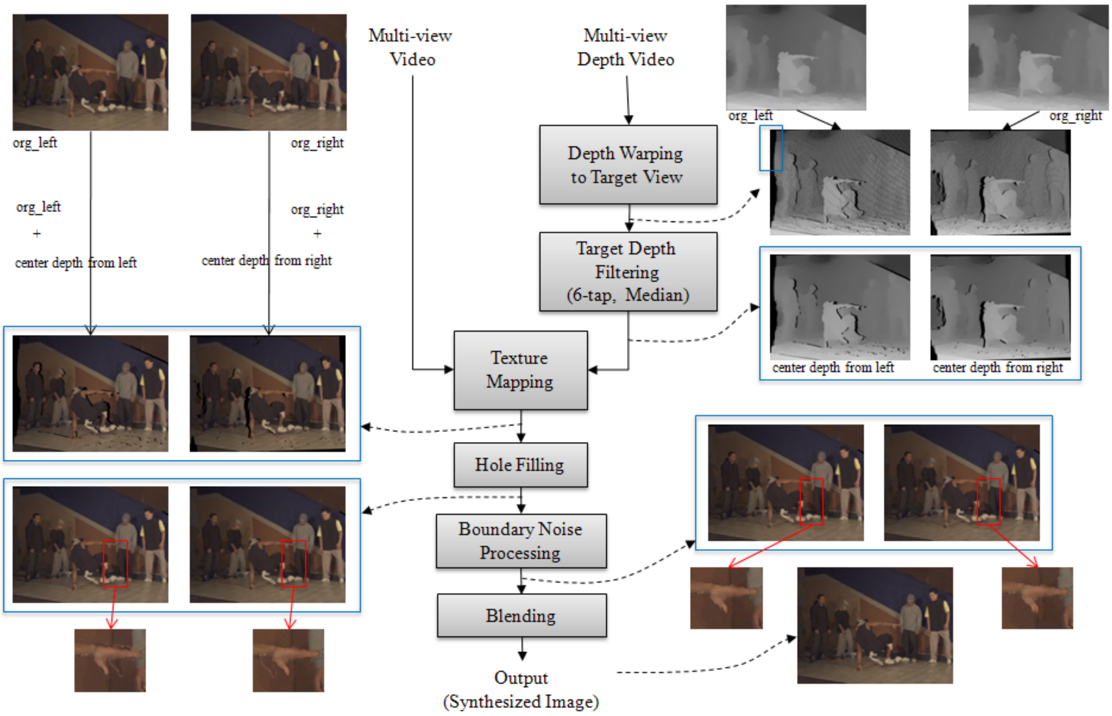


Fig. 1. Block Diagram of the View Synthesis

2.1. Pixel Correspondence using 3D warping

In order to map every pixel in the reference image, we define the pixel correspondence using 3D warping. As illustrated in Fig. 2, a position **X** in the real world is projected to **x** and **x'**. In other words, **x** and **x'** are corresponding pixels. If we know the depth of a pixel **x** in the reference image, we can find the corresponding pixel **x'.** More mathematical relationship is described in the document of boundary filtering [2].

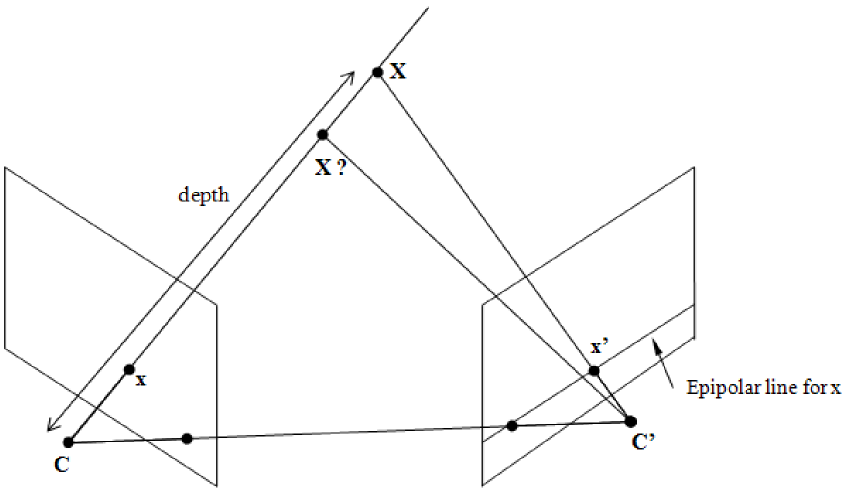


Fig. 2. Point Correspondence Geometry

2.2. Depth Map Warping

Instead of using direct pixel mapping, we warped the input depth map into the target viewpoint. Figure 3 shows the warped depth maps. The middle images are the warped depth map which contains many holes in black. In order to reduce the holes, we classified the hole into two categories: small hole and large hole (or hole). Small hole is caused by the truncation error in the floating point calculation. Those small holes can be filled in with interpolation such as 6-tap filter or median filter. The bottom rightmost images in Fig. 3 are the final depth maps to be used for texture mapping.

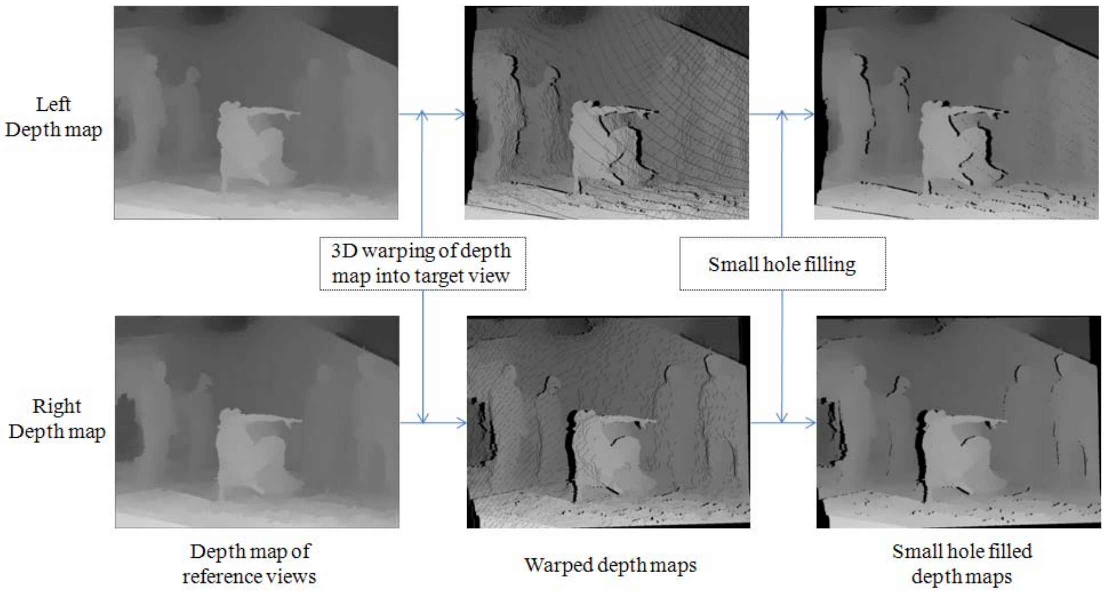


Fig. 3. Depth Map Warping

2.3. Texture Mapping

Using the 3D warping, we mapped all pixel correspondences between the reference and target viewpoint images. Next procedure is reconstruction an image at the target viewpoint uing warped depth map. Without the depth hole, all available pixels having depth data can get a color data using 3D warping. The result images are described in Fig. 4. The white areas are newly exposed regions so called ‘disocclusion’. If we have only one reference viewpoint image, we need to use inpainting technique to fill in the hole area. However, since 3DV system allows using any viewpoint data, we can exploit any viewpoint image. In our method, we use two neighboring reference images. The data of the newly exposed region can be found in the other side reference viewpoint image. For example, if the reference image is left view, the newly exposed area reveals at the right side of an object. It means that the alternative data exists at the right viewpoint image. More explanation will be followed in the next subsection.

Fig. 4. Synthesized Images using Warped Depth Maps

2.4. Hole Filling

When we change the viewpoint, some areas are disappeared and some are newly exposed. The disappeared area is useless in the target view, but the newly exposed area is revealed region at the target viewpoint, but there is no information in the reference image. If we have only one viewpoint image and its depth map, we may use inpainting method to fill in the area. 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.

In order to find the information of the hole area, we need to detect the hole region to be filled in. We detect it by checking the synthesized image having hole. In Figure 4, we colored the hole are in white. Next, we find the alternative texture value referring to the other reference view. Figure 5 explains about the hole filling method.

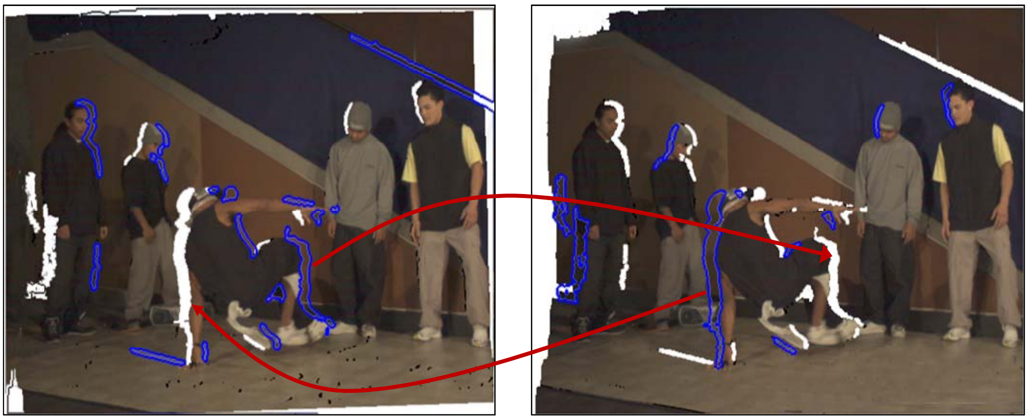


Fig. 5. Hole Filling

As shown in Fig. 5, the hole region is appeared at the right side of objects when we warp the left view into the right virtual view. The corresponding texture information of its hole region can be found at the right view. The blue contour of the left image depicts the corresponding texture of hole area, and the red arrow indicates the pair of alternative texture information and hole. We can fill in the hole area by copying the textures from the other view.

2.5. Boundary Noise Filtering

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. 6, fractions of the thumb exist at 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 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.

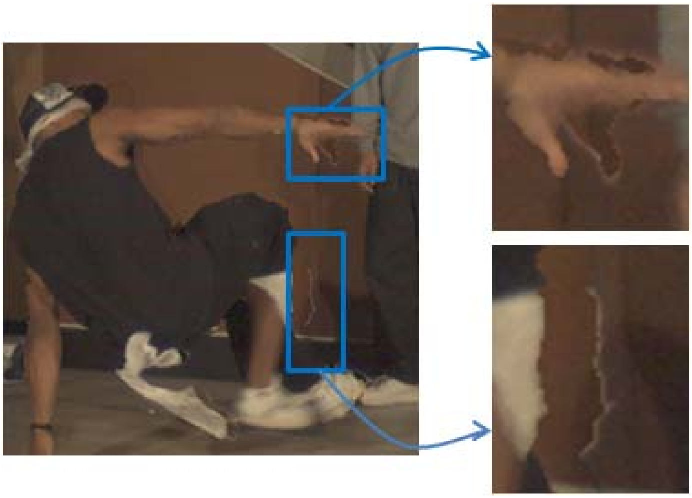


Fig. 6. Boundary Noise

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 process of boundary filtering is described in the Fig. 7.

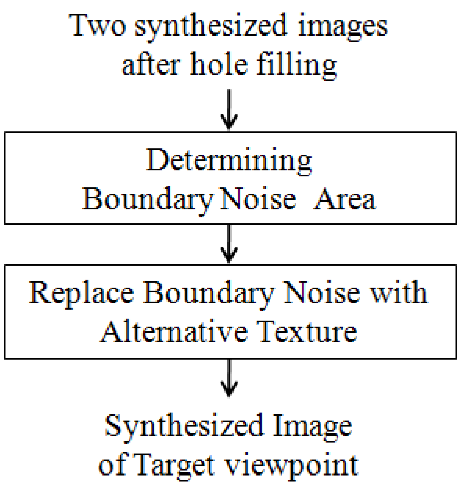


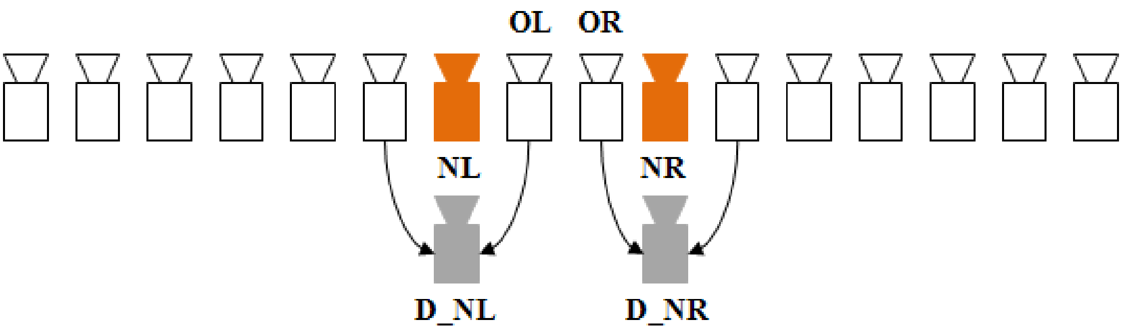
Fig. 7. Process of Boundary Noise Filtering

3. Experimental Results

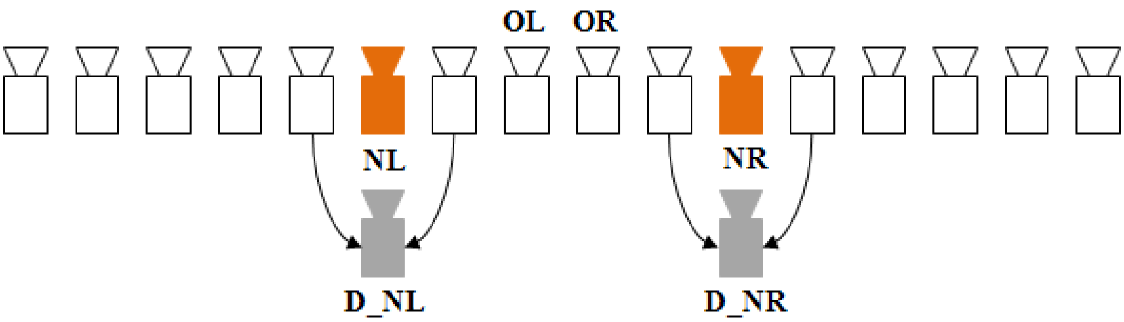
3.1. Depth Estimation

3.1.1. Selection of Reference Views for Depth Estimation

Since MPEG 3DV experts agreed with using the depth estimation software provided by Nagoya University, we also used it to generate depth maps. We followed the conditions of experiment described in the output document [3]. Figure 8 describes the procedure of generating a depth map. For the narrow baseline case, we chose two nearest views as described in Fig. 8(a). In medium baseline case, we chose the reference view the second nearest views as Fig. 8(b).



(a) Narrow Baseline Case



(a) Medium Baseline Case

Fig.8. Depth Map Generation

The version of depth estimation SW is 'DepthEstimation 1.1' which is updated August 31, 2008. We fixed the bug in the depth estimation tool that fails to set the data type. The tool has 6 different parameters for depth estimation. Table 1 is the parameter set we have used. We chose six 3DV test sequences: 'Book_arrival', 'Leaving_laptop', 'Door_flowers', 'Newspaper', 'Pantomime', and 'Dog'. The total length of each sequence is 100.

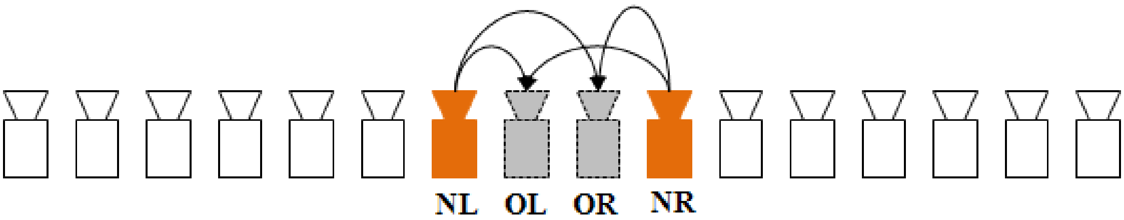
Table 1. Parameters of Depth Map Estimation and View Synthesis

Smoothing Coefficient	Baseline Basis	Precision	Search Level	Filter	Matching Method
5.0	Minimum	Integer	Integer	Bi-linear	Conventional

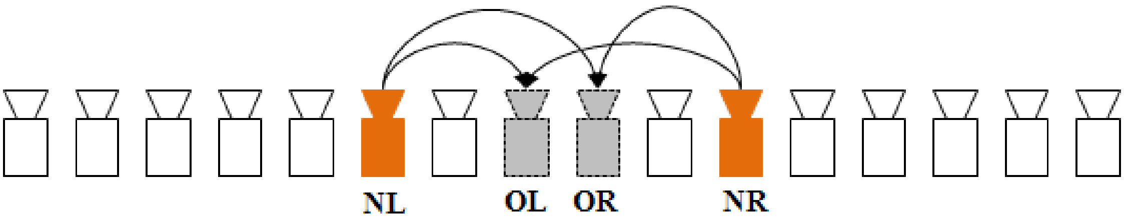
3.2. View Synthesis

3.2.1. Configuration of View Synthesis

The configuration of view synthesis for experiment is followed by the document of "Description of Exploration Experiments in 3D Video Coding". The document directs two baselines: narrow and medium. As illustrated in Fig. 9, We used two reference views, NL and NR, and synthesized two intermediate viewpoint images, OL and OR. In addition, we used following depth parameters such as 'Znear' and 'Zfar' as presented in Table 2, 3. Total number of frames synthesized is 100 for each sequences.



(b) Narrow Baseline Case



(b) Medium Baseline Case

Fig. 9. Configuration of View Synthesis

Table 2. Depth ranges of narrow baseline

Narrow Baseline	Left View		Right View	
	Znear	Zfar	Znear	Zfar
Book_arrival	24.876	49.752	24.705	49.410
Leaving_laptop	24.876	49.752	24.705	49.410
Door_flowers	22.112	49.752	21.960	49.410
Newspaper	3393.977	7542.171	3393.977	7542.171
Pantomime	3907.726	8221.651	3907.726	8221.651
Dog	5297.545	8701.977	3095.592	9241.909

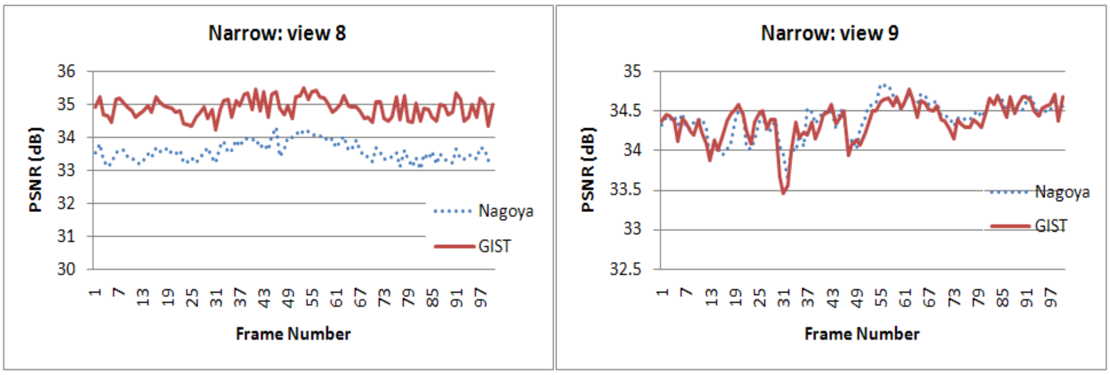
Table 3. Depth ranges of medium baseline

Medium Baseline	Left View		Right View	
	Znear	Zfar	Znear	Zfar
Book_arrival	25.480	50.960	24.705	49.410
Leaving_laptop	25.480	50.960	24.705	49.410

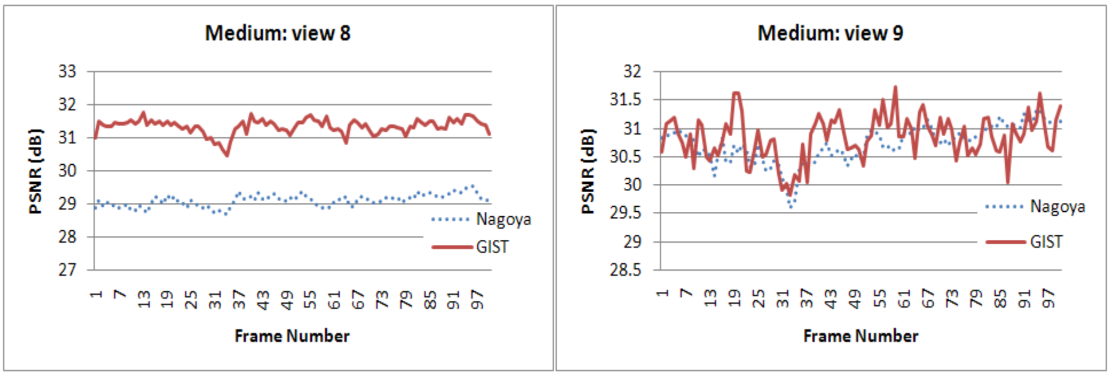
Door_flowers	11.697	51.173	21.960	49.410
Newspaper	3393.977	7542.171	3393.977	7542.171
Pantomime	3907.726	8221.651	3907.726	8221.651
Dog	5297.545	8701.977	5297.545	8701.977

3.2.2. Results of View Synthesis

In order to evaluate quality of synthesized images, we set the Nagoya's view synthesis tool to a reference SW. Since the target views are the left and the right, we compared both separately. From Fig. 10 to 15, all results are described by graphs. Table 4 ~ 9 show the average PSNR values for each sequence



(a) Narrow Baseline

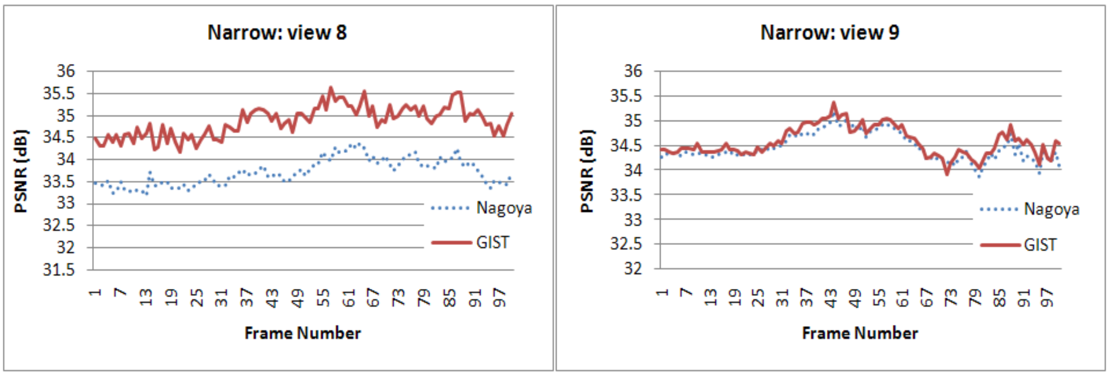


(b) Medium Baseline

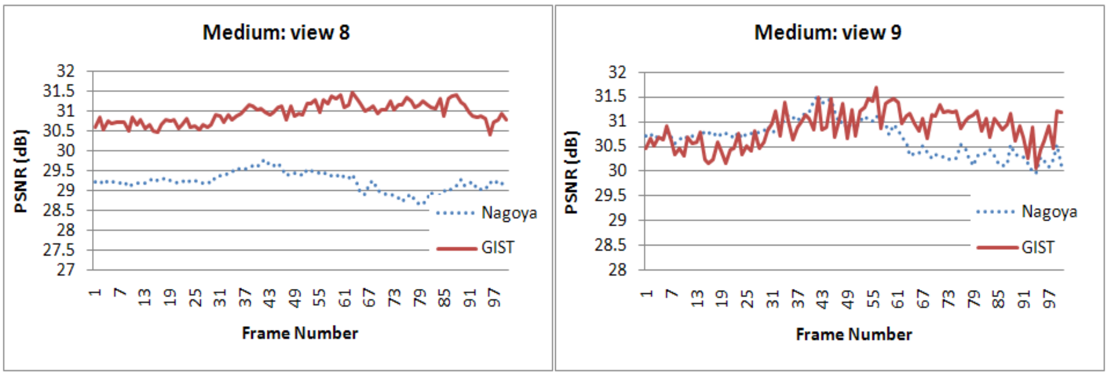
Fig. 10. Results on 'Book_arrival'

Table. 4. Average PSNR of 'Book_arrival'

	Narrow Baseline		Medium Baseline	
	View 8	View 9	View 8	View 9
Nagoya	33.5767	34.3998	29.1222	30.7358
GIST	34.9040	34.3786	31.3728	30.8444



(a) Narrow Baseline

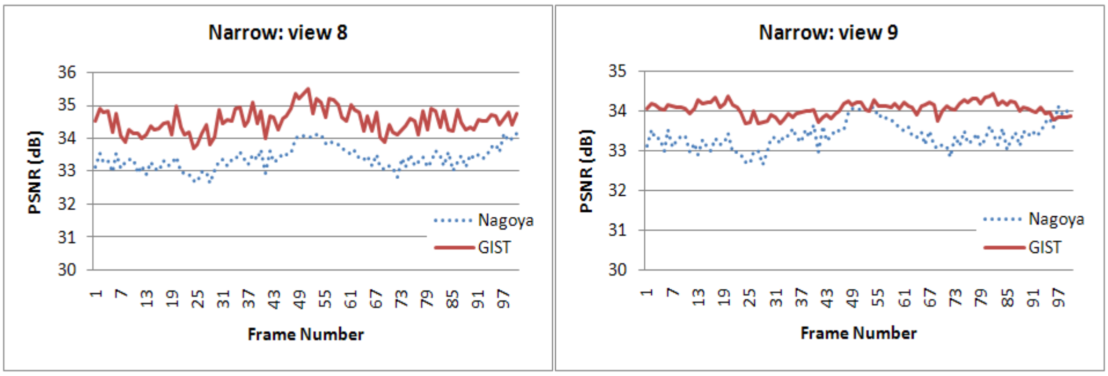


(a) Medium Baseline

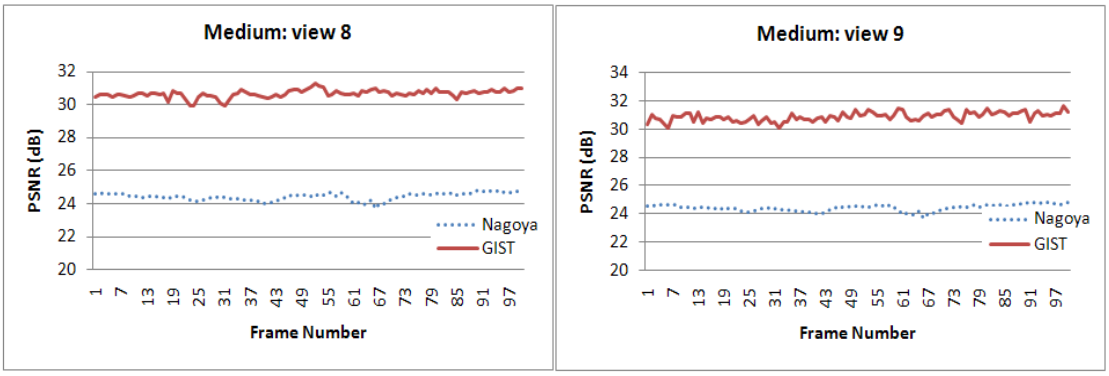
Fig. 11. Results on 'Leaving_laptop'

Table. 5. Average PSNR of 'Leaving_laptop'

	Narrow Baseline		Medium Baseline	
	View 8	View 9	View 8	View 9
Nagoya	33.7224	34.4818	29.2287	30.6687
GIST	34.8626	34.5868	30.9438	30.8690



(a) Narrow Baseline

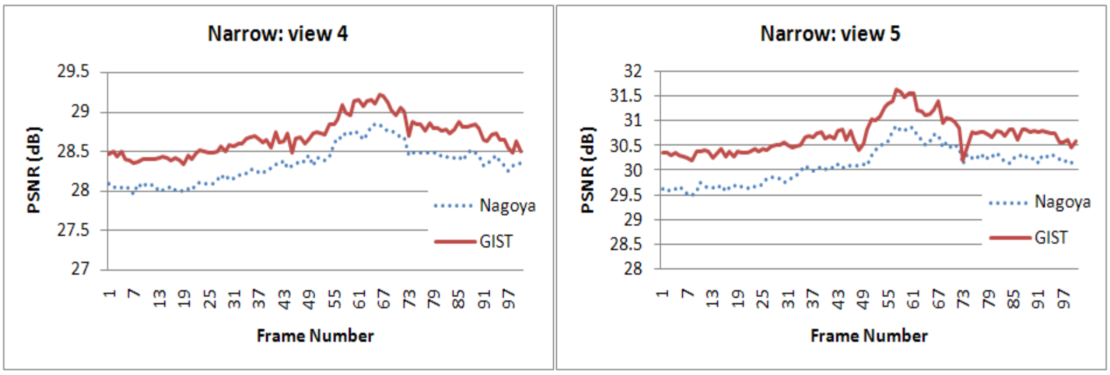


(a) Medium Baseline

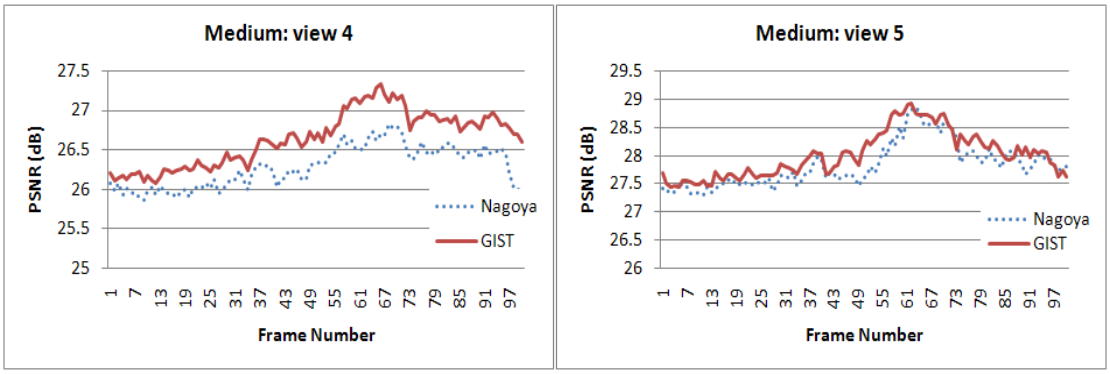
Fig. 12. Results on ‘Door_flowers’

Table. 6. Average PSNR of ‘Door_flowers’

	Narrow Baseline		Medium Baseline	
	View 8	View 9	View 8	View 9
Nagoya	33.3994	33.3994	24.4205	24.4205
GIST	34.5257	34.0598	30.6746	30.8605



(a) Narrow Baseline

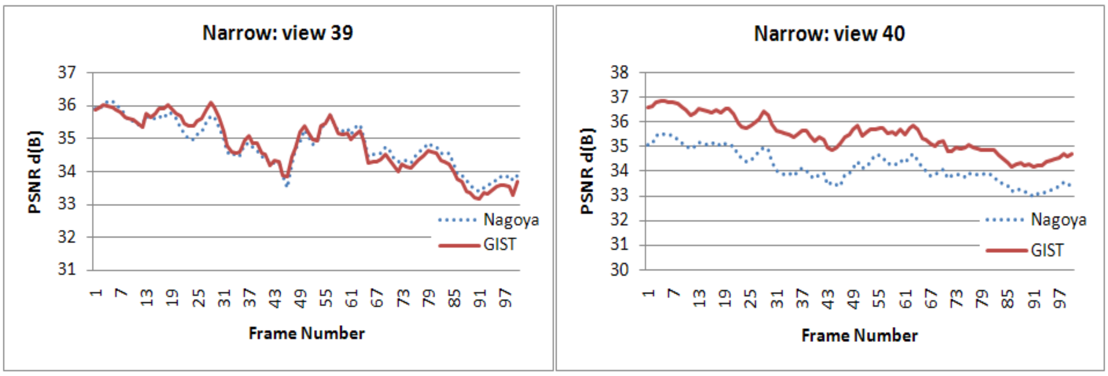


(a) Medium Baseline

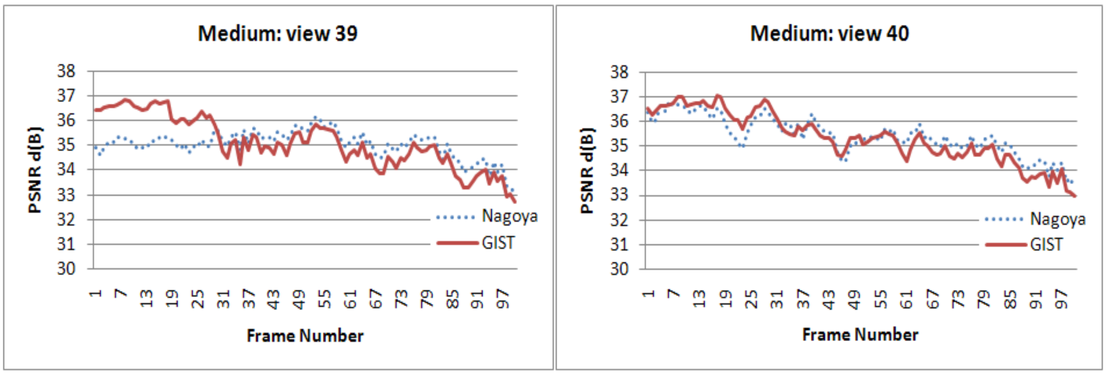
Fig. 13. Results on ‘Newspaper’

Table. 7. Average PSNR of ‘Newspaper’

	Narrow Baseline		Medium Baseline	
	View 8	View 9	View 8	View 9
Nagoya	28.3397	30.1229	26.2859	27.8345
GIST	28.6909	30.7112	26.6493	28.0170



(a) Narrow Baseline

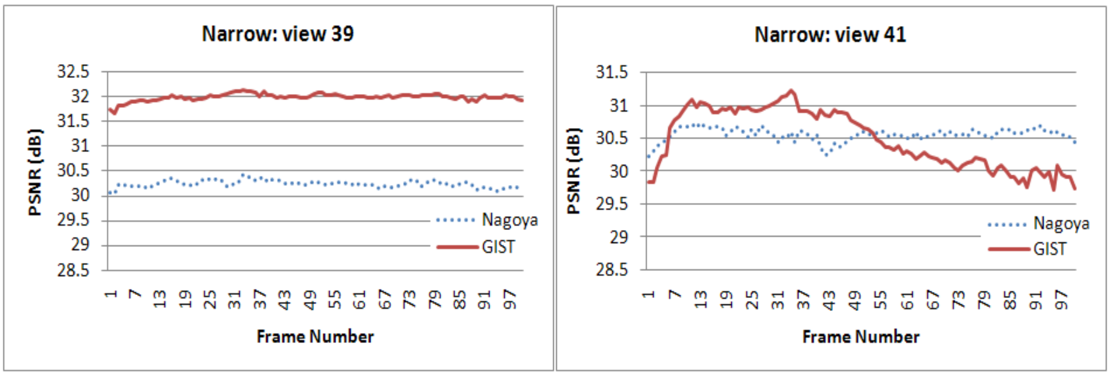


(a) Medium Baseline

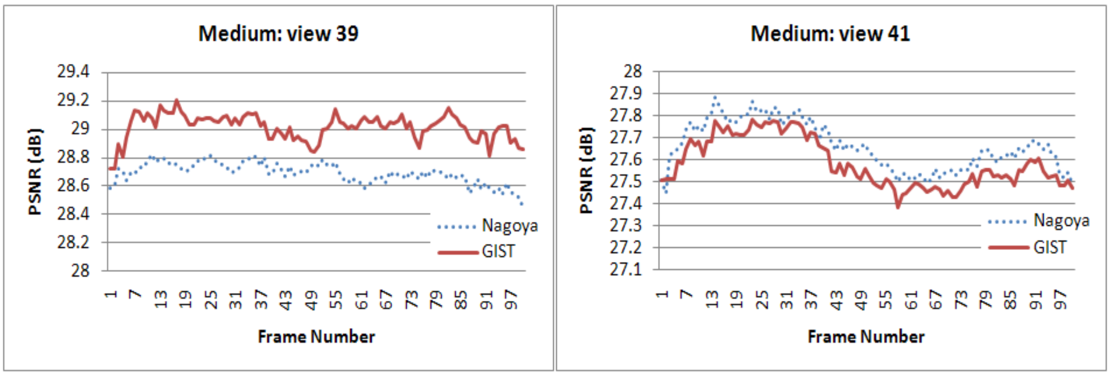
Fig. 14. Results on 'Pantomime'

Table. 8. Average PSNR of 'Pantomime'

	Narrow Baseline		Medium Baseline	
	View 8	View 9	View 8	View 9
Nagoya	34.8284	34.1958	35.0406	35.4043
GIST	34.8096	35.4792	35.1226	35.3084



(a) Narrow Baseline



(a) Medium Baseline

Fig. 15. Results on ‘Dog’

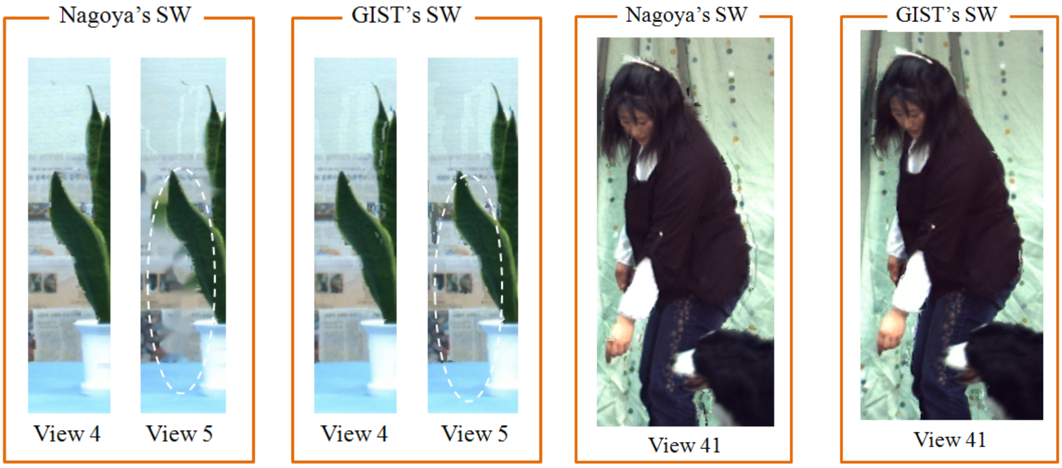
Table. 9. Average PSNR of ‘Dog’

	Narrow Baseline		Medium Baseline	
	View 8	View 9	View 8	View 9
Nagoya	30.2446	30.5592	28.6872	27.6617
GIST	31.9843	30.4937	29.0138	27.5859

In HHI’s sequences, all views are improved average PSNR compared with Nagoya’s SW. However, average PSNR values of right view on ‘Dog’ sequence are lower than that of results of Nagoya’s SW.

3.2.3. Results on Boundary Region

In the last meeting, we proposed a boundary noise processing eliminating boundary noises around depth discontinuity. As we mentioned above, we include the method in our view synthesis tool. Consequently, we obtained clean synthesized images if the sequences having large disparities. Following figures are the comparison of synthesized images. As shown in Fig. 16, our method generates better background image. This comes from using difference hole filling technique. To fill in the hole area, Nagoya’s SW uses inpainting technique, but we exploit the synthesized image referring to the other reference image. In other words, we just copy the background region for the hole area from the other synthesized image. Therefore, we get more clean background image.



(a) Synthesized Image on 'Newspaper' (b) Synthesized Image on 'Dog'

Fig. 16. Comparison for Boundary Noise Processing

4. Conclusion

We have explained about the view synthesis tools that we have implemented. Our tool generates an arbitrary viewpoint image around the hole area. In particular, we implemented the background noise processing extracting the boundary noise caused by inaccurate depth discontinuity. As a result of experiments, most of views are improved in PSNR. We suggest discussing results and updating plan for the view synthesis tool of Nagoya's during this Busan meeting.

5. Acknowledgements

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

6. References

[1] ISO/IEC JTC1/SC29/WG11 "Call for Contributions on FTV Test Material", N9468, October 2007.

[2] ISO/IEC JTC1/SC29/WG11 "Boundary Filtering on Synthesized Images for 3D Video," m15597, July 2008.

[3] ISO/IEC JTC1/SC29/WG11 "Description of Exploration Experiments in 3D Video Coding," N9991, July 2008.