Conversion of free-viewpoint 3D multi-view video for stereoscopic displays

Citation for published version (APA):

Do, Q. L., Zinger, S., & With, de, P. H. N. (2010). Conversion of free-viewpoint 3D multi-view video for stereoscopic displays. In *Proceedings of the 2010 IEEE International Conference on Multimedia and Expo (ICME), 19-23 July 2010, Singapore* (pp. 1730-1734). Institute of Electrical and Electronics Engineers. <https://doi.org/10.1109/ICME.2010.5583175>

DOI:

[10.1109/ICME.2010.5583175](https://doi.org/10.1109/ICME.2010.5583175)

Document status and date:

Published: 01/01/2010

Document Version:

Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

CONVERSION OF FREE-VIEWPOINT 3D MULTI-VIEW VIDEO FOR STEREOSCOPIC DISPLAYS

Luat Do¹, Svitlana Zinger¹, and Peter H. N. de With^{1,2}

¹ Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, Netherlands

² Cyclomedia Technology B.V., P.O. Box 68, 4180 BB Waardenburg, The Netherlands

Email: {Q.L.Do, S.Zinger, P.H.N.de.With}@tue.nl

ABSTRACT

This paper presents our ongoing research on view synthesis of free-viewpoint 3D multi-view video for 3DTV. With the emerging breakthrough of stereoscopic 3DTV, we have extended a reference free-viewpoint rendering algorithm to generate stereoscopic views. Two similar solutions for converting free-viewpoint 3D multi-view video into a stereoscopic vision have been developed. These solutions take into account the complexity of the algorithms by exploiting the redundancy in stereo images, since we aim at a real-time hardware implementation. Both solutions are based on applying a horizontal shift instead of double execution of the reference free-viewpoint rendering algorithm for stereo generation (FVP stereo generation), so that the rendering time can be reduced by as much as 30–40 %. The trade-off however, is that the rendering quality is 0.5–0.9 dB lower than when applying FVP stereo generation. Our results show that stereoscopic views can be efficiently generated from 3D multi-view video by using unique properties in stereoscopic views, such as identical orientation, similarities in textures and small baseline.

Index Terms—three-dimensional television (3DTV), free-viewpoint interpolation, Depth Image Based Rendering (DIBR), stereoscopic view generation.

1. INTRODUCTION

Three-dimensional television (3DTV) at high resolution is likely to be the succeeding step after the broad acceptance of HDTV. The introduction of depth signals along with texture videos enables rendering views from different angles. This technique is called Depth Image Based Rendering (DIBR) and is a popular research topic in recent years. One attractive feature of DIBR is *Free-Viewpoint* (FVP) [1], where the user chooses the view position from which he would like to watch a video. To enable free-viewpoint, we assume that we have several input video streams captured by multi-view cameras and each stream consists of a texture and depth signal. In the European iGlance project [2], a combination of the above-mentioned technologies is pursued for developing a real-time FVP 3DTV receiver. One of the main goals of this project is the development of a state-of-the-art FVP rendering algorithm. Taking into account the emerging breakthrough of stereoscopic screens, we extend

this reference FVP rendering algorithm to create stereoscopic vision, by rendering left and right views for the user, thus enabling a 3D viewing experience. For this purpose, we have developed two methods for generating stereoscopic views from multi-view video using the reference FVP rendering algorithm. Recent research shows that stereoscopic views can be generated from various types of video signals. Zhang *et al.* [3, 4] employ monocular video with an additional depth signal to synthesize virtual stereo images. Knorr *et al.* [5] generate stereo images from a 2D video sequence with camera movement. An approach using omnidirectional cameras is developed by Yamaguchi *et al.* [6] and Hori *et al.* [7] for creating separate views for each eye. However, none of these methods utilize multi-view video and they are therefore not applicable to our situation. Our starting point is a free-viewpoint 3D system configuration, from which there are possibilities to create a stereo signal. As multi-view processing in 3D is inherently expensive, we aim at developing options with a low complexity which is suited for a real-time implementation. Evidently, the solutions should have a sufficiently high quality.

In Section 2, we briefly introduce the reference FVP rendering algorithm. In Section 3, we describe the two methods we have developed for generating stereoscopic views using the reference FVP algorithm. In Section 4, these two methods are evaluated and in the last section, conclusions and recommendations are presented.

2. VIEW SYNTHESIS ALGORITHM

In this section, we explain the reference FVP rendering algorithm, which is used for generating stereoscopic views in the next section. The principal steps of this FVP rendering algorithm are depicted in Fig. 1 and will now briefly be described. A more detailed description can be found in [8]. In the first step, a virtual view is created by projecting or warping from the two nearest cameras to a user-defined position. The second step closes *cracks* and holes that are caused by view projection. Then the two projected images are blended and in the last step, the remaining disocclusions are inpainted. This FVP rendering algorithm is similar to [3, 9, 10, 11] but it has three distinguishing properties.

- Ghosting artifacts are reduced by omitting the areas at

edges between foreground and background from projecting to the virtual view position.

- A median filter is employed to close holes that are created by projection of one view to another.
- The quality of disocclusion inpainting is increased by taking into account the depth information at the edges of the disoccluded areas.

Fig. 1. Sequence of the principal steps in the reference FVP rendering algorithm.

In the next section, we present two methods for generating stereoscopic views by extending the reference FVP rendering algorithm and trying to limit the required amount of operations.

3. CONVERTING MULTI-VIEW VIDEO TO STEREOSCOPIC VIEWS

Let us now study the generation of stereoscopic views from 3D multi-view video. Both solutions are based on generating a virtual view using the reference FVP rendering algorithm described in the previous section. In general, stereo images can be generated by applying this algorithm twice: one time for each channel of the stereo signal. However, this leads to the double amount of operations compared to generating a single view. To minimize the computational effort, we propose two solutions. In the first solution, the second view is not generated with the reference FVP rendering algorithm but instead, it is created by *horizontal shifting* the virtual view to the right. In this case, the shift is proportional to the baseline of the stereoscopic vision. A drawback of this method is that the horizontal shifting produces large disocclusions, from which it is known that they cause very annoying artifacts [9, 8]. The second solution combats the large disocclusions by first generating a virtual view with the reference FVP rendering algorithm and subsequently performing a horizontal shift to the left and right of this virtual viewpoint for obtaining stereoscopic views. In this way, the disocclusions are divided between the two views. We will explain the horizontal

shifting with a brief example. In Fig. 2, a pair of stereo images is depicted. It can be seen that the orientation of the two images is identical. From [12], we know that the warping of one image to another is described by Equation (1):

$$\lambda_2 p_2 = K_2 R_2 [K_1 R_1]^{-1} \lambda_1 p_1 + K_2 (t_2 - t_1), \quad (1)$$

where K_n , and R_n , for $n \in [1, 2]$, are the intrinsic camera parameters and the rotation matrices, respectively, from the stereo images. Since we have a pair of stereo images, the K_n and R_n matrices for both the left and right views are equal. The values λ_1 and λ_2 denote the relative depth of an object to the viewpoint. Since the left and right views have an identical orientation, λ_1 and λ_2 are equal. The translations t_1 and t_2 describe the relative position or offset of the viewpoint in relation to the absolute xyz -coordinate system. Because the two viewpoints have an identical orientation, the difference $t_2 - t_1$ is only non-zero in the x -direction. When we apply the above observations,

Fig. 2. Orientation of a pair of stereo images.

we can rewrite Equation (1) as $\lambda p_2 = \lambda p_1 + K(t_2 - t_1)$, which can be worked out to

$$\lambda \begin{pmatrix} x_2 \\ y_2 \\ 1 \end{pmatrix} = \lambda \begin{pmatrix} x_1 \\ y_1 \\ 1 \end{pmatrix} + \begin{bmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} \Delta x \\ 0 \\ 0 \end{pmatrix},$$

which simplifies to

$$\begin{pmatrix} x_2 \\ y_2 \end{pmatrix} = \begin{pmatrix} x_1 + \frac{f}{\lambda} \Delta x \\ y_1 \end{pmatrix}. \quad (2)$$

The coordinates of the left view are represented by x_1 and y_1 , and x_2 and y_2 represent the projected coordinates at the right view. We can clearly see that the horizontal shift operator does not change the y -position. Furthermore, the x -position is shifted by an amount proportional to the baseline (Δx) of the stereo pair and the depth value (λ) of the projected coordinate.

The primary advantage of performing a horizontal shift compared to normal projection is the low complexity of computation. This is clearly seen when we compare Equation (1) with Equation (2). From the latter equation, we note that the displacement calculation of the x -coordinate involves only one addition, one multiplication and one division. Furthermore, because of the identical orientation of the viewpoints in the stereo image pair, the displacement of the y -coordinate is always zero.

3.1. Method 1: generating right stereo image from a virtual left stereo image

The first solution for generating stereoscopic views from multi-view video involves the interpolation of one virtual view (the left view) with the reference FVP rendering algorithm and the creation of a second virtual view by horizontal shifting to the right. Fig. 3 depicts the diagram and stepwise visualization of our first method for generating stereo images from multi-view video. First, we use the reference FVP rendering algorithm to create a virtual left image. From this image, we apply a horizontal shift to the right-hand side to generate a right stereo image. As mentioned earlier, the horizontal shifting produces disocclusions which appear at the right stereo image. The disoccluded regions are located at the right-hand side of foreground objects and can be filled in by a two-step approach (*inpainting by projection*) as follows,

1. We determine the depth value of every disoccluded pixel by searching for the nearest background in a circular direction.
2. The disoccluded areas are inpainted by projecting to the right reference camera, driven by the depth values found in the first step. We inpaint only the disoccluded pixels that have a similar depth value as the warped depth value at the right reference image.

After these two steps, there still may be remaining disocclusions due to the geometry of the scene. These areas can then be inpainted by the method used in the reference FVP rendering algorithm (*FVP inpainting*).

3.2. Method 2: generating left and right stereo images from a virtual image in between

The largest drawback of the previously described method is the considerable amount of disocclusions, which is proportional to the length of the baseline. One way to reduce this problem is to generate a virtual viewpoint between the left and right stereo

Fig. 3. Method 1: generating right stereo image from a virtual left stereo image.

images and perform a horizontal shift to the left and right-hand side to create two stereo images. The disocclusions will then be divided between the two virtual stereo images, resulting in the spreading of disocclusion artifacts over the two images. From the authors disocclusion inpainting research in [8], we expect that this method of generating stereo vision from multi-view video should give less annoying artifacts than our first method. Fig. 4 depicts our second solution for stereo conversion. First,

Fig. 4. Method 2: generating left and right stereo image from a virtual image in between.

we generate a virtual image between the positions of the left and right stereo images with the reference FVP rendering algorithm. Then we perform a horizontal shift to the left and right from this virtual image by applying Equation (2) of Section 3. The disocclusions produced by horizontal shifting are inpainted by projection and the remaining disocclusions are filled in by FVP inpainting. It should be noticed that now the baseline (Δx) is half of the baseline of the previous solution and thus the displacement of the x -coordinate for each stereo image is reduced by a factor of two as well. Comparing the second method with the first method, we can say that we have reduced the large disocclusions and spread the disocclusions over two images to gain a higher image quality, by applying two times a horizontal shift. In the next section we analyze each processing step of Fig. 3 and Fig. 4, and evaluate the overall performance of Method 1 and 2, where both are compared to applying twice the reference FVP rendering algorithm for stereo generation (FVP stereo generation).

4. STEREO IMAGE QUALITY ASSESSMENT

In order to evaluate the quality of the images generated by the algorithms presented in this article, we perform two series of experiments, where we measure the PSNR (Peak Signal-to-Noise Ratio) between these images and the “ground truth” and the timings of each algorithm. We synthesize a 3D model that contains three cubes of different sizes rotating in a room with paintings on its walls. Having this model allows us to obtain the ground truth that can be compared to our stereo image generation re-

sults. We define a viewpoint for which a couple of stereo images has to be generated. We employ two cameras - one from each side - around this viewpoint. The angle between these two cameras is 8° . Fig. 5 presents the PSNR for 100 frames for the following three cases:

- the right image of the stereo pair is generated by the reference FVP rendering algorithm and compared to ground truth; the resulting curve ('FVP') gives the best performance;
- the right image is generated by performing a horizontal shift from the left image as described by Method 1. This produces the worst results in terms of PSNR ('Method 1');
- the right image is obtained by horizontal shifting from a virtual image in between the stereo images as described by Method 2 ('Method 2'). The performance in terms of PSNR is better than Method 1.

Fig. 5. PSNR for the right images of stereo pairs generated by FVP and two methods described above.

We present the quality measured for the right image because it is the worst-case scenario for our algorithms. We can observe in Fig. 5 that Method 2 from Section 3.2 provides better results compared to the single shift approach (Method 1). This is explained by the reduced number of disoccluded areas, since we do have a virtual image in between and only need to perform small shifts to obtain the left and right stereo images. The generated disocclusions form an intensity shadow at the right-hand side of the cubes in Fig. 6 and 7. This intensity shadow is significantly reduced at the same locations in Fig. 7.

Let us now investigate in more detail the performances of Method 1 and 2, and compare those to FVP stereo generation. The average results of 100 frames for these three algorithms are summarized in Table 1. The second and fourth row show the rendering time of each algorithm for both the left and right stereo images, implemented in MATLAB. The fifth row indicates the percentage of disocclusions from the total image prior to FVP inpainting for the left and right stereo image and the last row consists of the average PSNR for the left and right stereo image generated by the three methods. From Table 1 we conclude that generating stereoscopic views from an FVP image

Fig. 6. The synthetic scene used for algorithm performance evaluation: shadows at the right-hand side of the cubes are disocclusions produced by Method 1.

Fig. 7. The synthetic scene used for algorithm performance evaluation: disoccluded areas at the right-hand side of the cubes are reduced by Method 2.

can be performed very efficiently. We note that Method 1 is about 40% faster than FVP stereo generation but loses 1.3 dB in rendering quality for the right stereo image. Using Method 2, the loss in rendering quality can be reduced to an average of 0.6 dB compared to FVP stereo generation. However, since we have applied horizontal shifting twice, the rendering time of Method 2 is only 30% smaller than that of FVP stereo generation. The efficiency of Method 1 and 2 can be explained by three aspects. First, the horizontal shifting operator only computes the displacement of the x -coordinates. Second, median filtering is not used since horizontal shifting produces almost no holes or cracks. Third, the remaining disoccluded areas of Method 1 and 2 are 2–5 times smaller than those of FVP stereo generation.

5. CONCLUSIONS

Our study is directly applicable to free-viewpoint stereoscopic vision with recent 3D screens. Such viewing will provide a stereo pair of images for a viewpoint chosen by the user. Generating stereo images from multi-view video with texture and

	FVP stereo		Method 1		Method 2	
Rend. L+R (s)	2.82		1.73		1.90	
	left	right	left	right	left	right
Rendering (s)	1.40	1.42	1.40	0.33	0.27	0.23
% occlusions	0.76	0.69	0.76	0.32	0.20	0.13
PSNR (dB)	33.0	32.9	33.0	31.6	32.3	32.5

Table 1. Summary of performance of the two stereo generation methods compared to FVP stereo generation.

depth signals is a challenging problem, especially when computational cost should be low to obtain an efficient hardware implementation. In this paper, we have presented two ways of stereo image generation from 3D multi-view video that avoid a double execution of the reference FVP rendering algorithm. This reduces the amount of required operations, but the quality of the results is a concern. We have evaluated this quality in terms of PSNR and found that generating a virtual image in the middle of the stereo pair position and shifting it to the left and right-hand side (Method 2) provides the better performance. By measuring the rendering duration of the two methods for creating stereo images, we have observed that the rendering time for Method 1 and 2 can be reduced by 40% and 30%, respectively, compared to FVP stereo generation. However, the reduction in rendering time comes with a trade-off. On the average, the rendering quality of Method 1 and 2 is 0.7 dB and 0.6 dB, respectively, lower than that of FVP stereo generation.

Our evaluation of the two methods for stereo generation from multi-view shows us that it is possible to exploit the redundancy in stereo images for developing highly efficient stereo generation algorithms. However, further research is needed in order to make a well-grounded choice between Method 1 and 2. An interesting question is the subjective stereo-experience of users when we compare Method 1 with Method 2 using real-life multi-view video.

6. REFERENCES

- [1] M. Tanimoto, "FTV (free viewpoint television) for 3D scene reproduction and creation," in *CVPRW '06: Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop*, Washington, DC, USA, 2006, p. 172, IEEE Computer Society.
- [2] S. Zinger, D. Ruijters, and P. H. N. de With, "iGLANCE project: free-viewpoint 3D video," in *17th International Conference on Computer Graphics, Visualization and Computer Vision (WSCG)*, 2009.
- [3] C. Fehn, "Depth-image-based rendering (DIBR), compression, and transmission for a new approach on 3D-TV," in *Stereoscopic Displays and Virtual Reality Systems XI.*, May 2004, vol. 5291, pp. 93–104.
- [4] L. Zhang and W. J. Tam, "Stereoscopic image generation

based on depth images for 3D TV," *Broadcasting, IEEE Transactions on*, vol. 51, no. 2, pp. 191–199, 2005.

- [5] S. Knorr, M. Kunter, and T. Sikora, "Stereoscopic 3D from 2D video with super-resolution capability," *Image Commun.*, vol. 23, no. 9, pp. 665–676, 2008.
- [6] K. Yamaguchi, H. Takemura, K. Yamazawa, and N. Yokoya, "Real-time generation and presentation of view-dependent binocular stereo images using a sequence of omnidirectional images," *Pattern Recognition, International Conference on*, vol. 4, pp. 4589, 2000.
- [7] M. Hori, M. Kanbara, and N. Yokoya, "Novel stereoscopic view generation by image-based rendering coordinated with depth information," in *SCIA*, 2007, pp. 193–202.
- [8] L. Do, S. Zinger, and P.H.N. de With, "Quality improving techniques for free-viewpoint DIBR," in *Stereoscopic displays and applications XXII*, 2010.
- [9] C. L. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-quality video view interpolation using a layered representation," in *ACM SIGGRAPH 2004 Papers*, New York, NY, USA, 2004, pp. 600–608, ACM.
- [10] A. Smolic, K. Müller, K. Dix, P. Merkle, P. Kauff, and T. Wiegand, "Intermediate view interpolation based on multiview video plus depth for advanced 3D video systems," in *ICIP*. 2008, pp. 2448–2451, IEEE.
- [11] Y. Mori, N. Fukushima, T. Yendo, T. Fujii, and M. Tanimoto, "View generation with 3D warping using depth information for FTV," *Image Commun.*, vol. 24, no. 1-2, pp. 65–72, 2009.
- [12] L. McMillan, Jr., *An image-based approach to three-dimensional computer graphics*, Ph.D. thesis, Chapel Hill, NC, USA, 1997.