VLSI Design of View Synthesis for 3DVC/FTV

Jongwoo Bae' and Jinsoo Cho[2],

[1] Department of Information and Communication Engineering, Myongji University
San 38-2 Namdong, Cheoin-Gu, Yongin-Si, Gyeonggi-Do 449-728, Korea
jwbae@mju.ac.kr

[2] Department of Computer Engineering, Gachon University
San 65, Bokjung-Dong, Sujung-Gu, Sungnam City, Gyeonggi-Do 461-701, Korea
jscho@gachon.ac.kr

Abstract. In this paper, we propose a VLSI design of view-synthesis architecture for 3DVC/FTV. After the compressed video stream is decoded, the intermediate view point images are created by view-synthesis method using depth-image-based rendering (DIBR) to produce more view points. We propose a novel VLSI architecture to implement DIBR. We demonstrate that the proposed architecture can handle 80 frames per second for full-HD video. The proposed design is 12K gates, and runs at 172.1MHz in TSMC LVT90 process.

Keywords: 3DVC, FTV, VLSI, view synthesis, depth-image-based rendering.

1 Introduction

The demand for supporting various video codec standards is rapidly increasing in consumer electronics markets such as D-TV and mobile devices. 3D display is introduced to the D-TV market and many TVs are manufactured as a 3DTV. MPEG and many other video specialist groups have worked on the standardization of the 3D video codec. Compression based on the prediction between the multiple views, i.e., multi-view video codec (MVC) has been studied recently. Likewise, Free view point TV (FTV) is actively studied to produce the intermediate view images between the existing view images employing one of the 2D to 3D image conversion technique called Depth-Image-Based Rendering (DIBR). This technique requires a lot of computation for the generation of the intermediate view images [1-6].

In this paper, we propose a VLSI design for the view synthesis of 3DVC/FTV using DIBR. For the multi-view computation of HD video, the amount of computation increases drastically as the number of view points, image resolutions, and frame rates increase. Compared with the traditional 2D video codec, the circuits need to be designed to support

[2] Dr. Cho is the corresponding author.

higher performance and bandwidth. There are contributions still yet to be made for the VLSI design of the view synthesis of 3DVC/FTV.

This paper is organized as follows. Section 2 introduces the background of the 3D video technology we are working on. Section 3 proposes a detailed explanation of the VLSI architecture for view-synthesis of 3DVC/FTV using DIBR. Section 4 summarizes the design results. Finally, section 5 concludes the paper.

2 Proposed Method

2.1 Background

For the design of 3DVC/FTV, we perform view synthesis after the video decoding [1-6]. Intermediate images are synthesized from two left and right images. We can keep producing additional view images by view synthesis. After we obtain the decoded images and the depth maps for left and right views, we synthesize the intermediate image from them as follows. The left and right view images make the intermediate image by warping. The intermediate image made from the left view image contains holes caused by the information unavailable from the left view. The intermediate image made from the right view has holes for the same reason. The two intermediate images are supposed to be identical because they are made for the same view point. Therefore, they are the same images with different hole information. Merge is a process to create a combined intermediate image from two different intermediate images of left and right views. The intermediate image obtained from merge can still have defects due to two reasons: warping errors and holes. Holes can exist because of the absence of certain view information. Hole-filling is a process to eliminate the holes in the image after merge. After hole-filling, we have the final intermediate image ready to display [1-6].

2.2 Proposed Architecture

The top architecture of view synthesis consists of image warping, merge, and hole-filling modules. The intermediate result of each module is stored in the dual buffer SRAM as a FIFO architecture to increase the throughput. Therefore, the whole design operates in a pipelined fashion. View controller controls the whole operation. Since the first slice of the image is not pipelined, the view controller handles it separately. The address generation module is designed to fetch the pixel data before and after the X coordinate position of the current pixel for hole-filling.

Image warping is the most critical part in creating the intermediate image. The 3D image is represented as a 2D image, i.e., the 3D coordinate of (x, y, z) is transformed into the 2D coordinate of (x, y). The depth map provides the z-value to the 2D image. Based on it, the x-value of the 2D image is adjusted properly to form a new view point image.

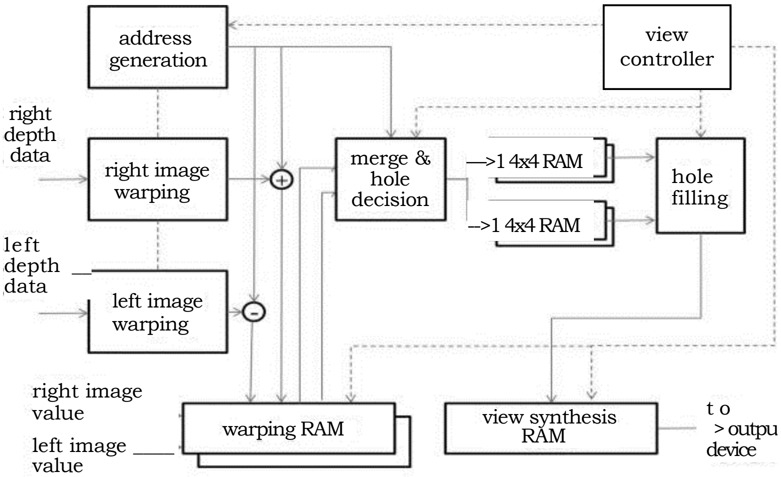

Fig. 1. Top architecture of view synthesis

Merge combines the two intermediate images of a view into one image. Because they are supposed to be same images, the pixels of the same position must be identical. If they are not identical, either some of them have errors during the warping, or holes are present. When one of them results in being a hole, the pixel of the other image is copied into the hole. If none of them are holes, then the average value of the two different pixels is used. After merge, hole-filling assigns values to the remaining holes. For each hole, the average of the left and right pixels are computed and assigned to the hole. The VLSI design for merge and hole-filling consists of the control logic and simple arithmetic.

The design shown in Fig. 1 is a pipelined architecture using dual buffer FIFOs between the modules. If we use a non-dual buffer FIFO, the design becomes sequential, and we can save the reduced RAM area at the cost of performance reduction.

3 Experimental results

We designed two different architectures: a sequential, and a pipelined. TSMC LVT90 process is used for the synthesis. Table I shows the experimental results of the two designs. The frame size used for the test is HD 1920x1080 pixels. For the sequential design, the operating frequency is 51.2MHz due to a long critical path. The size of the design is 10K gates, which does not greatly differ from that of a pipelined design. The performance of a sequential design is only 8 frames per second. The sequential design cannot handle the HD video in real-time.

The pipelined design can operate at 172.1MHz. The size of the design is 12K gates. The pipelined design's performance is 10 times higher than that of the sequential design, which can process 80 frames per second. The requirement for full HD video display is 60 frames per second. The pipelined design can meet the requirement of full HD video.

Table 1. Experimental results of the three proposed designs.

Design	Operating frequency (MHz)	Gate count (K gates)	Performance (frames/sec)
sequential	51.2	10	8
pipelined	172.1	12	80

4 Conclusion

In this paper, we presented the VLSI design of view-synthesis for 3DVC/FTV. The design of view-synthesis using DIBR consists of warping, merge, and hole-filling. Two architectures were designed and compared for the paper: a sequential, and a pipelined. The top architecture of a pipelined design was shown in detail. The pipelined architecture could provide the performance enough to handle full-HD video.

Acknowledgements

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (grant number 2011-0026386).

References

1. MPEG, Report on Experimental Framework for 3D Video Coding. In ISO/IEC JTC1/SC29/WG11, MPEG2010/N11631 (2010)
2. Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification. In ITU-T Rec. H.264 I ISO/IEC 14496-10 AVC (2003)
3. C. Lee and Y. Ho: View Synthesis using Depth Map for 3D Video. In APSIPA (2009)
4. Y. Mori, et. al.: View generation with 3D warping using depth information for FTV. In Signal Processing: Image Communication. Vol. 24. Issue 1-265-72 (2009)
5. C. Fehn: Depth-Image-Based Rendering (DIBR), compression and transmission for a new approach on 3D-TV. In Proc. SPIE Stereoscopic Displays and Virtual Reality Systems XI, pp. 93-104 (2004)
6. D. Tian, et. al.: View synthesis techniques for 3D video. In Proc. SPIE 2009 (2009)