VIEW-DEPENDENT CODING OF LIGHT FIELDS BASED ON FREE-VIEWPOINT IMAGE SYNTHESIS

Yuichi Taguchi and Takeshi Naemura

Graduate School of Information Science and Technology, The University of Tokyo
7-3-1, Hongo, Bunkyo-ku, Tokyo, 113-8656, Japan
{yuichi, naemura}@hc.ic.i.u-tokyo.ac.jp

ABSTRACT

This paper proposes a view-dependent light field coding scheme using some image-based rendering techniques prior to coding. The proposed coder first synthesizes an image at a given viewpoint, which is called a *representative viewpoint*, and then predicts all input images by using the synthesized image as a reference. It can produce a view-dependent scalable bitstream. This means that the quality of synthesized views around the representative viewpoint is kept high even at extremely low bit rates, and the quality of views away from there is improved according to the increase of the bit rate. Our experimental results show that this coding scheme also achieves good coding efficiency for both multi-camera images and integral photography, which are common light field representations.

***Index Terms*—** Image coding, Data compression, Rendering (computer graphics), Prediction methods

1. INTRODUCTION

Image-based rendering (IBR) techniques have attracted a lot of research interest since they have a great potential for synthesizing photorealistic 3D scenes. IBR data sets, such as light fields [1], are often constructed from multi-view images captured with an array of cameras [2, 3] or lenslets [4]. For high quality image synthesis, however, hundreds or thousands of images are necessary. Consequently, efficient coding schemes are required to transmit or store such large amount of image data.

A number of light field compression schemes have been developed in recent years [5, 6]. Basic methods for this purpose include the vector quantization (VQ) [1] and the disparity-compensated prediction (DCP) [7]. If 3D geometry information is available, the coding efficiency can be increased considerably [8, 9]. While predictive methods typically improve the coding efficiency by exploiting the inter-image correlation, they introduce dependencies between images, which restrict random access to the data to render a novel view [10].

These techniques are commonly designed to compress a light field uniformly. Therefore the ease of random access and the reconstruction quality of a synthesized image are approximately constant regardless of the viewpoint. However, there are many applications in which a certain viewpoint image is significant and required fast decoding. For example, when we interactively browse synthesized views over a network, the data for our current view is more important than that for the other views. If the current viewpoint image is only transmitted, however, we can not change the viewpoint immediately due to the network latency. For enabling us to change the viewpoint before the arrival time of the next frame data, the data for the neighboring views should be transmitted with increasing bandwidth.

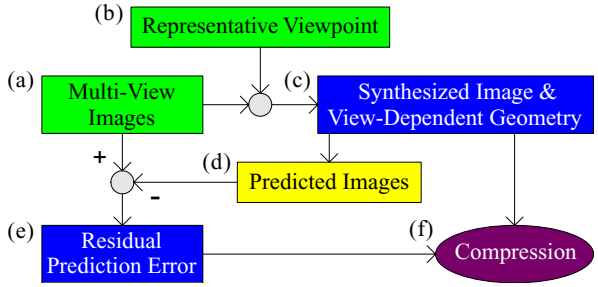


Fig. 1. A block diagram of the proposed coding scheme.

This paper proposes a scalable light field coder based on free-viewpoint image synthesis. We call it a view-dependent coder because it places priority on a given viewpoint, which is called a *representative viewpoint*. The proposed coder first synthesizes an image at a representative viewpoint, and encodes the light field data by using the synthesized image as key information. It produces a view-dependent scalable bitstream, with which the representative viewpoint image is decoded first, and the quality of views around the representative viewpoint is kept higher than that of other views at low bit rates. The quality of views away from the representative viewpoint is improved according to the increase of the bit rate. Our experimental results show that this coder also achieves good coding efficiency for both multi-camera images and integral photography.

2. VIEW-DEPENDENT CODER

2.1. Coding Procedure

Shown in Fig. 1 is a block diagram of the proposed coder. First of all, we synthesize an image at a representative viewpoint using some image-based rendering techniques. A view-dependent geometry model is also estimated during this image synthesis. Secondly, all of the input images are predicted by using the synthesized image and the estimated geometry model. The residual prediction error is generated if the quality of the predicted image is not sufficient. Finally, the synthesized image, view-dependent geometry model, and residual prediction errors are compressed and stored into a hierarchical bitstream shown in Fig. 2.

This prediction process is similar to that used in the model-aided predictive coding [8], but we take a novel approach that uses a synthesized image at an arbitrary viewpoint as a reference image. Thus this coder provides both direct access to the representative viewpoint image, and good coding efficiency by using a predictive method with a view-dependent geometry model.

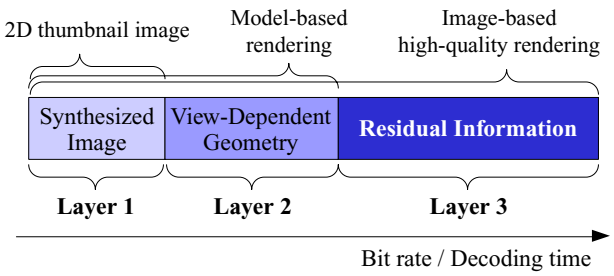


Fig. 2. A hierarchical structure of the encoded bitstream.

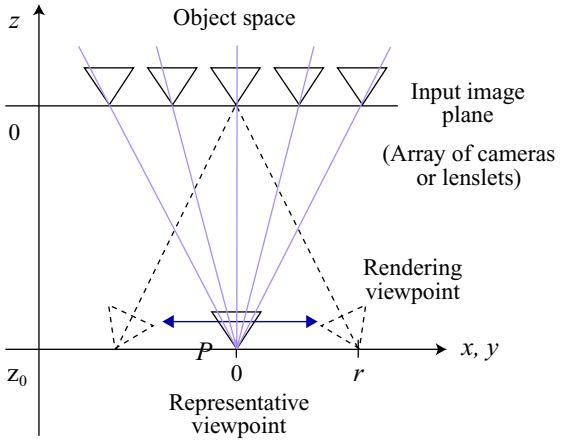


Fig. 3. An evaluation method for the synthesized image.

2.2. Hierarchical Bitstream

Our coder produces a hierarchical bitstream shown in Fig. 2. This bitstream can be used in three classes of applications depending on the bit rate or decoding time as follows.

The layer 1 is a synthesized image at a representative viewpoint. This can be a *thumbnail* of the light field because we can see an overview of the 3D scene. The layer 2 includes a view-dependent geometry model. Using layers 1 and 2, we can synthesize novel views by model-based rendering techniques. However, the quality of views away from the representative viewpoint might not be high enough due to the geometry errors and occlusions. The residual information is stored in the layer 3. Using all the layers, we can render high quality views with some interpolation methods like the light field rendering [1].

Thus our coder provides the view-dependent scalability. If the bit rate of the residual data in the layer 3 increases, the quality of views away from the representative viewpoint could be improved. In addition, our coder can also achieve the compatibility with conventional 2D image formats by using the layer 1 image as a base image and the data of the layers 2 and 3 as its extension information. For example, we created a JPEG-compatible bitstream in [11].

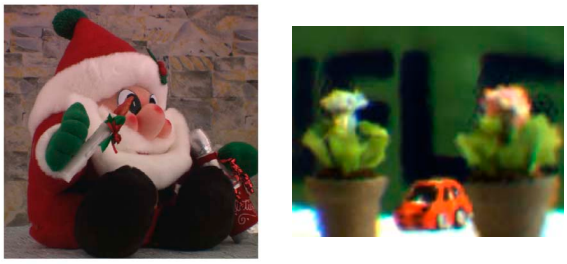
2.3. Evaluation Method

We typically evaluate the coding performance of a light field coder by reconstructing the input multi-view images and measuring their quality. This means that the quality of the synthesized image is not evaluated directly. In this paper, we evaluate the proposed coder with the synthesized images as well as the normal measurements

(a) Doll — Multi-camera images (9 x 9 views of 640 x 480 pixels)

(b) Car and Flowers — Integral photography (26 x 23 views of 31 x 31 pixels)

Fig. 4. A part of input multi-view image set.



(a) Doll (b) Car and Flowers

Fig. 5. Synthesized images at the representative viewpoint.

with the input images. Such evaluation is essential because the proposed coder provides view-dependent reconstruction quality of the synthesize image by using a representative viewpoint image as a key frame.

Shown in Fig. 3 is our evaluation method for the synthesized image. We synthesize images using the original and compressed light fields, and measure the reconstruction quality by comparing these two images. The measurement is repeated for changing the rendering viewpoint whose distance from the representative viewpoint (P) is denoted as r. At low bit rates, the reconstruction quality would be lower when the distance r becomes longer.

3. IMPLEMENTATIONS

We implemented the proposed coder for two different data sets shown in Fig. 4: (a) *Doll* and (b) *Car and Flowers*, which are an example of multi-camera images and integral photography, respectively. The *Doll* image set provided by "the multiview image database courtesy of University of Tsukuba, Japan" consists of 81 (9×9) images of 640×480 pixels. Recording camera positions lie in a plane and are arranged in a regular grid. On the other hand, the *Car and Flowers* image set is created from an integral photography captured with our real-time IBR system named LIFLET [4], which employs an array of lenslets and an XGA camera. An integral photography consists of a set of small circle images, and each image is called an elemental image. We used 598 (26×23) elemental images of 31×31 pixels as the image set [1].

The proposed coder first reconstructs a view-dependent geometry model of the scene at the representative viewpoint, and synthe-

[1] Invalid pixels among the elemental images, i.e. the exterior portion of circular region, were padded with the nearest valid pixel color in order to reduce the high-frequency components.

sizes a novel image. In this implementation, we used a modeling and rendering algorithm described in [12] for multi-camera images, and [13] for integral photography. The representative viewpoint was set behind the center of the input images' plane. Shown in Fig. 5 is the synthesized image at this viewpoint, which is clear and sharp in the whole area due to the geometry model.

The synthesized image was once encoded using a standard block-DCT scheme, and then locally decoded to be used as a reference image. The input multi-view images were predicted by warping this reference with the geometry model. Since the prediction accuracy varied widely, a coding mode was selected for each macroblock of 16×16 pixels from the following:

- Only predicted
- Predicted and residual coded
- Intra-coded

If the predicted macroblock met a preset minimum reconstruction quality q_{min}, the "only predicted" mode was selected and no further information for this macroblock was coded. Otherwise the coder compared the evaluated value between the residual prediction error and the original input macroblock, and decided which should be coded. The "intra-coded" mode was selected if the evaluated value of the original image was better than that of the residual error. This decision process for the latter two modes is equivalent to that used in MPEG-2 Test Model 5 [14].

Finally, the residual errors were encoded using a block-DCT scheme as well as the synthesized reference image. The geometry model was losslessly compressed with a DPCM method. In the following experiments, those bits are taken into account for calculating the bit rate.

4. EXPERIMENTAL RESULTS

4.1. Rate-distortion Performance of the Input Images

Figure 6 illustrates the rate-distortion performance measured with the input images. The proposed coder was compared with two conventional coders: the JPEG coder encoded input images independently, while the MPEG-2 coder encoded them as a sequence of moving pictures. They are simple implementations of an intra-image coder and a DCP coder, respectively. The performance of the proposed coder was measured for changing the quality of residual information. This means that we changed the parameter q_{min}, which is a threshold value for selecting the macroblock modes, and the quantization factor of the residual information. The quality of the synthesized image and geometry model was kept constant through all measurements. We calculated the peak signal-to-noise ratio (PSNR) for each input image of the luminance value, and expressed its average and standard deviation as the reconstruction quality.

The proposed coder shows better performance than the other coders especially at low bit rates. The minimum bit rate of the proposed coder is much lower than that of the MPEG-2 coder since the geometry model introduces less overhead bits than the motion vectors for prediction. Though the MPEG-2 coder exceeds the proposed coder at high bit rates for the *Doll* image set, note that the MPEG-2 coder dose not have the view-dependent scalability. The performance of the proposed coder at high bit rates could be improved by using the predictive coding method between the residual errors.

It can also be seen that the performance of the MPEG-2 coder is worse than that of the JPEG coder for the *Car and Flowers* image set. Since the elemental images of the integral photography record a small part of the scene separately, they have less inter-image correlations than the multi-camera images. Therefore the prediction between the elemental images dose not work efficiently. However, the proposed coder shows good performance because it can achieve efficient prediction using the representative viewpoint image that records an overview of the scene as a reference (see Fig. 5(b)).

4.2. View-dependent Reconstruction Quality of the Synthesized Image

We evaluated the reconstruction quality of input images in Section 4.1. In this section, we evaluate the view-dependency of the quality for synthesized images using the method described in Section 2.3.

The experimental result is shown in Fig. 7. The reconstruction quality was measured at 36 viewpoints for each r, which is the distance between the representative viewpoint and the rendering viewpoint (see Fig. 3); these rendering viewpoints were placed on a circumference at regular intervals. The average and the standard deviation of the PSNR at these viewpoints are depicted against r. The bit rate of (A) to (H) corresponds to those in Fig. 6. At the rates (A) and (E), no residual information was used, i.e. the synthesized image and the geometry model were only used to reconstruct the synthesized image. The bit rate of the residual information increases from (B) and (F) to (D) and (H), respectively.

The view-dependency of the reconstruction quality can be observed for both data sets, that is, the PSNR value of the synthesized image decreases with increasing the distance r. The quality of views around the representative viewpoint is kept high even at low bit rates, and the quality of views away from the representative viewpoint is improved according to the increase of the residual bits. Our coder achieves the view-dependent scalability since it produces a hierarchical bitstream whose reconstruction quality is view-dependent as shown in this experiment.

5. CONCLUSIONS

In this paper, we proposed a view-dependent coding scheme based on free-viewpoint image synthesis. This coder produces a view-dependent scalable bitstream, with which the representative viewpoint image is decoded first and the synthesized view indicates view-dependent reconstruction quality. The experimental results showed that this coder also achieves efficient coding performance for two image sets captured by the different methods. Future work will be focused on extending this scheme to dynamic scenes, and developing a real-time streaming system of light fields.

Acknowledgement: We wish to acknowledge valuable discussions with Prof. H. Harashima and Mr. K. Takahashi at the University of Tokyo, Japan.

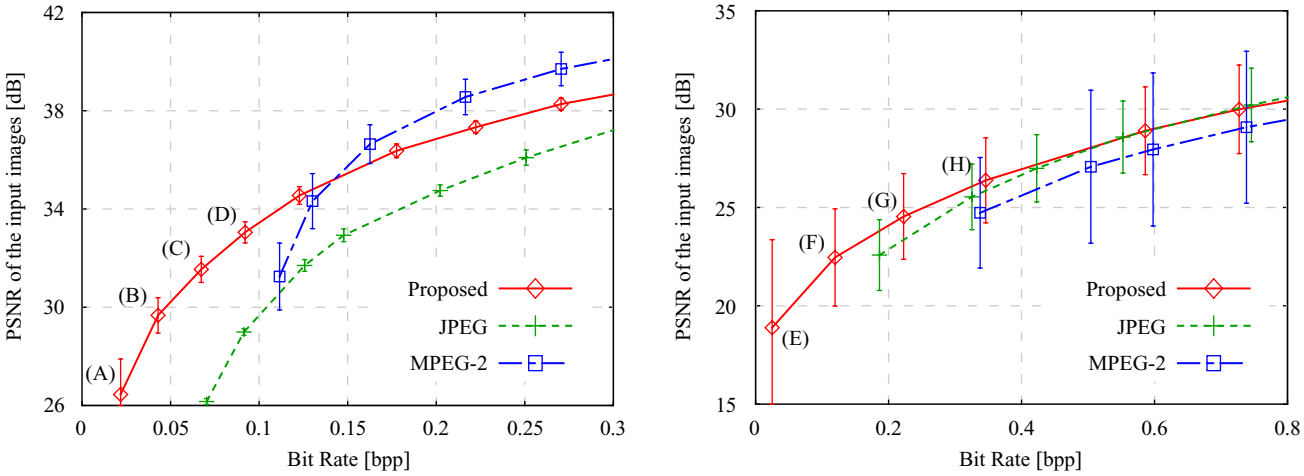
6. REFERENCES

[1] M. Levoy and P. Hanrahan, "Light field rendering," in *Proc. ACM SIGGRAPH'96*, Aug. 1996, pp. 31–42.

[2] B. Wilburn, N. Joshi, V. Vaish, E.-V. Talvala, E. Antunez, A. Barth, A. Adams, M. Horowitz, and M. Levoy, "High performance imaging using large camera arrays," in *Proc. ACM SIGGRAPH 2005*, July 2005, pp. 765–776.

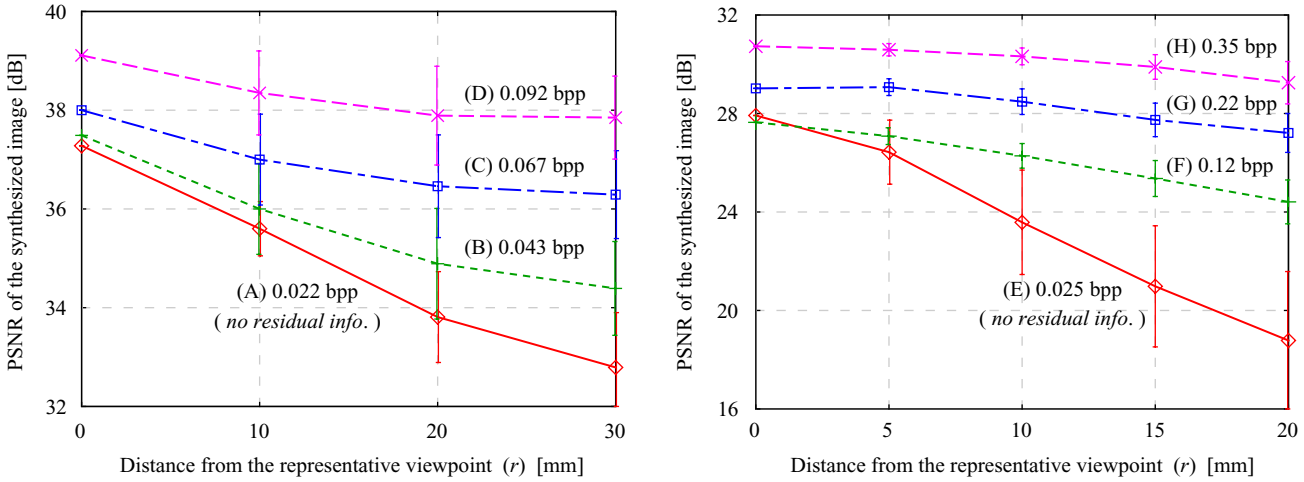
[3] M. Tanimoto, "FTV (free viewpoint television) creating ray-based image engineering," in *Proc. IEEE ICIP 2005*, Oct. 2005, vol. II, pp. 25–28.

[4] T. Yamamoto, M. Kojima, and T. Naemura, "LIFLET: Light field live with thousands of lenslets," ACM SIGGRAPH 2004 Emerging Technologies, etech_0130, Aug. 2004.



(a) Doll (b) Car and Flowers

Fig. 6. Rate-distortion curves of the input images.



(a) Doll (b) Car and Flowers

Fig. 7. Reconstruction quality of the synthesized image against the position of the rendering viewpoint.

[5] H.-Y. Shum, S. B. Kang, and S.-C. Chan, "Survey of image-based representations and compression techniques," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 11, pp. 1020–1037, Nov. 2003.

[6] A. Smolic and P. Kauff, "Interactive 3-D video representation and coding technologies," *Proc. IEEE*, vol. 93, no. 1, pp. 98–110, Jan. 2005.

[7] C. Zhang and J. Li, "Compression of lumigraph with multiple reference frame (MRF) prediction and just-in-time rendering," in *Proc. IEEE Data Compression Conference (DCC'00)*, Mar. 2000, pp. 253–262.

[8] M. Magnor, P. Ramanathan, and B. Girod, "Multi-view coding for image-based rendering using 3-D scene geometry," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 11, pp. 1092–1106, Nov. 2003.

[9] D. N. Wood, D. I. Azuma, K. Aldinger, B. Curless, T. Duchamp, D. H. Salesin, and W. Stuetzle, "Surface light fields for 3D photography," in *Proc. ACM SIGGRAPH 2000*, July 2000, pp. 287–296.

[10] P. Ramanathan and B. Girod, "Rate-distortion analysis of random access for compressed light fields," in *Proc. IEEE ICIP 2004*, Oct. 2004, pp. 2463–2466.

[11] Y. Taguchi and T. Naemura, "Free-viewpoint thumbnail for light field compression," ACM SIGGRAPH 2005 Posters #64, July 2005.

[12] K. Takahashi, A. Kubota, and T. Naemura, "A focus measure for light field rendering," in *Proc. IEEE ICIP 2004*, Oct. 2004, pp. 2475–2478.

[13] S. Mitsuda, T. Yamamoto, K. Takahashi, T. Naemura, and H. Harashima, "Interactive view synthesis from integral photography using estimated depth information," in *Proc. SPIE Three-Dimensional TV, Video, and Display II*, Sept. 2003, vol. 5243, pp. 116–124.

[14] "MPEG-2 TM5," http://www.mpeg.org/MPEG/MSSG/tm5/.