ON OBJECT-BASED COMPRESSION FOR A CLASS OF DYNAMIC IMAGE-BASED REPRESENTATIONS

Qing Wu, King-To Ng, Shing-Chow Chan and *Heung-Yeung Shum**
Department of Electrical and Electronic Engineering
The University of Hong Kong
{qingwu, ktng, scchan}@eee.hku.hk

* Microsoft Research, Asia
hshum@microsoft.com

ABSTRACT

An object-based compression scheme for a class of dynamic image-based representations called "plenoptic videos" (PVs) is studied in this paper. PVs are simplified dynamic light fields in which the videos are taken at regularly spaced locations along a line segment instead of a 2-D plane. To improve the rendering quality in scenes with large depth variations and support the functionalities at the object level for rendering, an object-based compression scheme is employed for the coding of PVs. Besides texture and shape information, the compression of geometry information in the form of depth maps is also supported. The proposed compression scheme exploits both the temporal and spatial redundancy among video object streams in the PV to achieve higher compression efficiency. Experimental results show that considerable improvements in coding performance are obtained for both synthetic and real scenes. Moreover, object-based functionalities such as rendering individual image-based objects are also illustrated.

1. INTRODUCTION

Image-based rendering (IBR) is a promising approach for the photo-realistic rendering of scenes and objects from a collection of densely sampled images. Since the data size associated with dynamic image-based representations is usually very large, efficient methods for its capturing, storage and transmission are active fields of research [1]. In [2], a multiple cameras system for capturing a class of dynamic image-based representation called "plenoptic videos" (PVs) was developed. It is a simplified light field for dynamic environments so that users can select their viewpoints on the camera plane of a linear video camera array. Using a parallel processing-based system, high-quality rendering of dynamic image-based representations using off-the-shelf equipment were obtained. Plenoptic videos are also closely related to multiview video sequences [3], [4], [1]. However, plenoptic videos usually rely on denser sampling in regular geometric configurations to improve the rendering quality. In addition, the random access to individual pixels in the compressed data stream, so-called the random access problem in IBR, becomes very important in real-time rendering.

In this paper, we study the object-based compression for the plenoptic video to facilitate its rendering, transmission and storage. Major advantages of the object-based representation are two-fold. Firstly, by properly segmenting image-based representations into objects at different depths, the rendering quality in large environments can be significantly improved. From plenoptic sampling [5], it was found that the spectral support of a light field is dependent on the depth values of the objects in the scene and no occlusions or depth discontinuities are assumed. However, for scenes with large depth variations, extremely high sampling would be required to overcome the rendering artifacts such as ghosting and blurring around depth discontinuities. One approach to overcome this problem is to segment the scene into depth layers so that the adverse effect of depth discontinuities can be mitigated. The idea has been demonstrated in the "pop-up light fields" [6], where excellent rendering quality could be achieved if the light field is properly segmented into layers of different depth values. In addition to this important advantage, by coding plenoptic videos at the object level, desirable functionalities such as scalability of image-based objects, error resilience, and interactivity with individual image-based objects, etc, can be achieved. For example, a compressed image-based object can be transmitted at a different rate and composited to different plenoptic videos at the receiver.

To support these object-based functionalities and improve the rendering quality, we propose to encode the texture, shape, grayscale alpha map (for matting) and depth information of each IBR object together. This scheme may be viewed as a generalization of our previous frame-based compression technique for plenoptic videos in [7], except that now arbitrarily shaped video objects rather than images with fixed size are encoded. The proposed method also shares many useful concepts with the MPEG-4 [8] video coding standard, however, additional geometry information in the form of depth maps have to be incorporated to facilitate the rendering of IBR objects. Moreover, the proposed compression scheme exploits both the temporal redundancy and spatial redundancy among video streams in the plenoptic video to achieve better compression efficiency.

In the proposed object-based framework, multiple video streams in the plenoptic video can be encoded into user-defined IBR objects, and reconstructed at the decoder for display and rendering at either the object level or frame level. A real-time system was constructed to demonstrate the proposed approach. Experimental results show that satisfactory rendering and composition results can be achieved.

The rest of the paper is organized as follows. A brief introduction to the plenoptic videos is given in Section 2. Section 3 discusses the proposed object-based compression scheme. Experimental results are presented in Section 4 and finally, conclusions are summarized in Section 5.

2. DYNAMIC IMAGE-BASED REPRESENTATIONS AND THE PLENOPTIC VIDEOS

0-7803-9134-9/05/$20.00 ©2005 IEEE

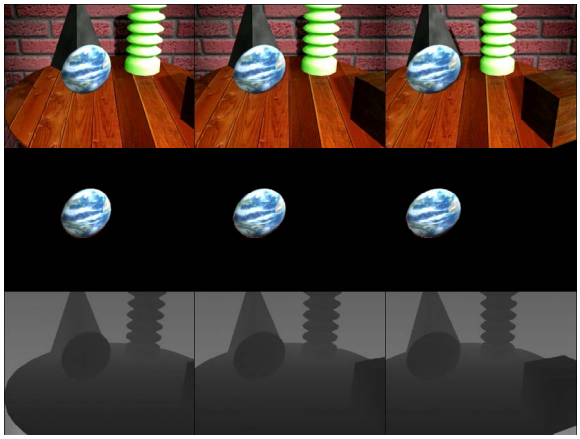

Fig. 1. Snapshots of (Top) synthetic PV "synthesis"; (Middle) the IBR object "ball"; (Bottom) the depth maps of the synthetic PV "synthesis".

Fig. 2. Snapshots of (Top) real-scene PV "dancer"; (Middle) the IBR object "dancer"; (Bottom) the depth maps of and the IBR object "dancer".

Central to IBR is the plenoptic function [9], which is a 7-dimensional (7D) function. Two important 4D static representations are the light fields [10] or lumigraphs [11] (lumigraph differs from light fields in using additional depth information), where densely sampled images taken on a 2D camera plane are used to render novel views or intermediate views. To further avoid the large dimensionality and the excessive hardware cost in capturing dynamic representations, a class of dynamic image-based representations called the simplified dynamic light field (SDLF) or the plenoptic video is proposed in [2], [7]. The plenoptic video is also a 4D plenoptic function and it is obtained by constraining the users' viewpoints along a line (or line segments) instead of a 2D plane in static light fields. Despite this simplification employed, plenoptic videos can still provide a continuum of viewpoints, significant parallax and lighting changes along line segments joining the camera arrays.

Figs. 1 and 2 show several snapshots from two plenoptic videos and IBR objects segmented from the scenes. In Fig. 1 is a synthetic sequence called "synthesis", while in Fig. 2 is a real-scene one called "dancer". The "ball" and the "dancer" in the scenes are segmented to form two IBR objects. A semi-automatic segmentation method called "lazy snapping" [12] is used to perform the segmentation. In the next section, the proposed object-based compression scheme for the coding of these IBR objects will be introduced.

3. THE PROPOSED OBJECT-BASED COMPRESSION SCHEME

After the IBR objects have been extracted, they can be compressed individually to provide functionalities such as scalability of contents, error resilience, and interactivity with individual IBR objects. For example, different IBR objects might be given different numbers of bits (and different amounts of channel coding) and hence different reconstruction qualities (error resilience). They might also be transmitted at different frame rates to achieve object scalability.

Fig. 4 shows the proposed method for the texture coding of an IBR object in the plenoptic videos. Only three video object (VO) streams are shown for simplicity, and we call them a group of video object field (GOVOF). In each VO stream, we have a view of the IBR object, which we shall refer to as the video object plane (VOP). There are two types of VO streams associated with each dynamic IBR object: *main* and *secondary* video object streams. Main VO streams are encoded similar to the MPEG-4 algorithm, which can be decoded without reference to other VO streams. For better performance, we also allow bi-directional prediction. To provide random access to individual VOP, we follow the structure of Group of VOP (GVOP) of MPEG-4 and employ it in the main stream. A GVOP contains an I-VOP and some P-VOPs and/or B-VOPs between two successive I-VOPs. I-VOPs are coded using intra-frame coding to provide random access point, while P-VOPs are coded by motion predictive coding using previous I- or P-VOPs as references. B-VOPs are coded by a similar method except that forward and backward motion compensations are performed by using nearby I- or P-VOPs as references, which are indicated by the block arrow in Fig. 4. The VOPs captured at the same time instant as the I-VOP in a main stream constitute an *I-VOP field*. Similarly, we define the *P-* and *B-VOP fields* as the VOP field containing respectively the P- and B-VOPs of the main VO stream. A VOP from the secondary stream in the I-VOP field are encoded using disparity-compensated prediction (DCP) or "*spatial prediction*" from the reference I-VOP in the I-VOP field. It is because adjacent light field images appear to be shifted relative to each other, similar to the effect of linear motion in video coding. This kind of prediction has been used in the coding of static light fields. Similarly, besides using temporal prediction in the same stream, the secondary P/B-VOPs also employ spatial prediction from their adjacent P/B-VOPs in the main stream for better performance.

Apart from texture images, each VO also contains depth maps and binary alpha maps and grayscale alpha maps. The latter one is useful in matting VOs during VO composition and rendering. Both grayscale alpha maps and depth maps are coded in a similar way as the luminance component of the texture, except that two pre-processing operations namely companding (the reciprocals of depth pixel values are actually coded) and scaling (scaled to an appropriate range for the encoder) are performed for depth maps.

Following MPEG-4, the shape of an object will be described using alpha maps. In a binary alpha map, each pixel is colocated with that of the luminance picture to indicate whether the current location is belonging to the object or not. Similar to MPEG-4,

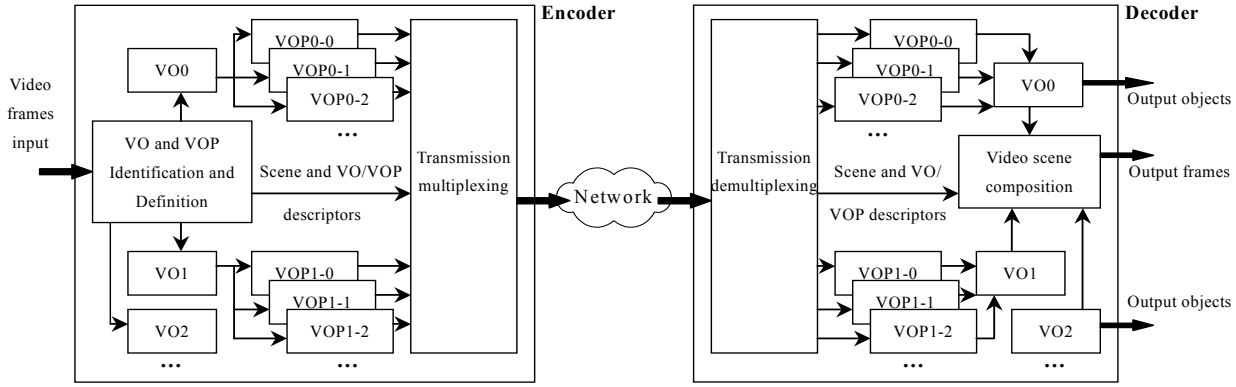

Fig. 3. Generic structure of the proposed object-based codec for the plenoptic video.

the context-based arithmetic encoding (CAE) algorithm [13] is employed to compress the binary shape information. Since intra-CAE codes shape information in intra mode, it is mainly used in encoding I-VOP in the main stream. In contrast, inter-CAE makes use of motion prediction from a shape mask reference, and therefore is used in other VOP types. An I-VOP in a secondary stream uses the VOP of the main stream as the shape mask reference for its shape coding, while a P-VOP or a B-VOP in a secondary stream may select a better shape mask reference from a VOP in the same secondary stream or the main stream. Macroblocks that lie within the object are coded similar to traditional video coders, while blocks at the boundary of an object can either be coded using padding or shape-adaptive DCT.

Fig. 3 shows the generic structure of our object-based codec. To perform object-based coding, different VOs in the scene are encoded separately. Each VO contains the VOPs distributed in all the streams in the plenoptic video, which are further defined with the corresponding shape, alpha maps and depth maps. Each VOP is then encoded based on its shape and motion information. The scene and VO/VOP descriptors for the plenoptic video are also encoded and multiplexed together with the VOPs, which are used to compose the video scenes at the decoder. The decoder can demultiplex and decode the VOPs for display or rendering.

4. EXPERIMENTAL RESULTS

To evaluate the performance of the proposed object-based compression scheme, a synthetic PV and a real-scene PV, both with the amount of 24 frames, are encoded respectively. The synthetic PV "synthesis" is produced by the 3D Studio Max software with a resolution of 320×240 pixels and 24-bit RGB components per pixel. The real-scene PV "dancer" has a resolution of 720×576 pixels in 24-bit RGB format, which were captured by our multiple video cameras system, as shown in Fig. 6. It consists of two linear arrays of cameras, each hosting 6 JVC DR-DVP9$_{AH}$ video cameras. The distance between two adjacent cameras is 15 cm, and the angle between the arrays can be flexibly adjusted. The corresponding depth maps are generated with 16 bits per pixel. Figs. 1 and 2 show a few snapshots of two PVs and two objects extracted from them – the olivary "ball" and the "dancer", respectively. The olivary ball with a complicated texture is rotating and moving from the left to the right, making the shape of ball vary violently during encoding. The depth maps of the scenes of synthetic PV and the depth maps of IBR object "dancer" are also shown. shown in Figs. 1 and 2, despite that the "synthesis" and "dancer" contain 9 and 6 streams, respectively. Figs. 7 and 8 show the Due to the space limitation, snapshots for only 3 streams are combined coding results with respect to PSNR in texture and shape coding for IBR objects "ball" and "dancer" at different bit rates achieved by using VM [14] rate control algorithm. The frame rates used for the PVs are 24 frames per second. For illustration, a Group of VOPs (GVOP) structure consisting of 12 VOPs (1 I-VOP, 3 P-VOPs and 8 B-VOPs) is employed. The curves denoted by "MPEG-4" represent the results using MPEG-4-like algorithm without spatial prediction, while those denoted by "SP-3", "SP-5" and "SP-7" represent the coding results using the proposed algorithm with 3, 5 and 7 VO streams within a GOVOF, respectively. It can be seen from Fig. 7 that, for the synthetic IBR object "ball", there is a considerable improvement in PSNR performance (4 dB) of the proposed object-based coding scheme over the direct application of the MPEG-4 to individual VO streams. The coding performances of SP-5 and SP-7 are slightly better than that of SP-3, while the former two are very close to each other. This is to be expected because when the disparity between two video streams increases, spatial prediction becomes less effective. The performance improvement for the real-scene IBR object "dancer", as shown in Fig. 8, is less significant compared with the synthetic sequence. This is mainly due to the slight position errors introduced by imperfect camera calibration, which destroys somewhat the correlation between the video streams. Therefore, the results for SP-3 and SP-5 are very close to each other. Since the variation in a depth map within an object is much less than texture information, it can be coded with a higher compression ratio than the latter. The rendering examples displayed in Fig. 5 are rendered using the reconstructed depth maps, where the average compression ratio of depth map for the IBR object "ball" is about 500 at a PSNR of 40 dB.

Finally we need to note that the performance of the proposed system can be further improved if other tools of MPEG-4 such as global motion vector, four motion vectors for a MB, direct prediction mode and so forth are incorporated in coding of the secondary VO streams. Moreover, it would be valuable to incorporate other advanced coding tools in the new H.264 standard into the proposed compression scheme for better coding performance. These will be studied in our future work. To further demonstrate the object-based functionality of the proposed codec, the renderings from the real-scene PV "dancer" at both the frame and the object levels are also shown in Fig.5.

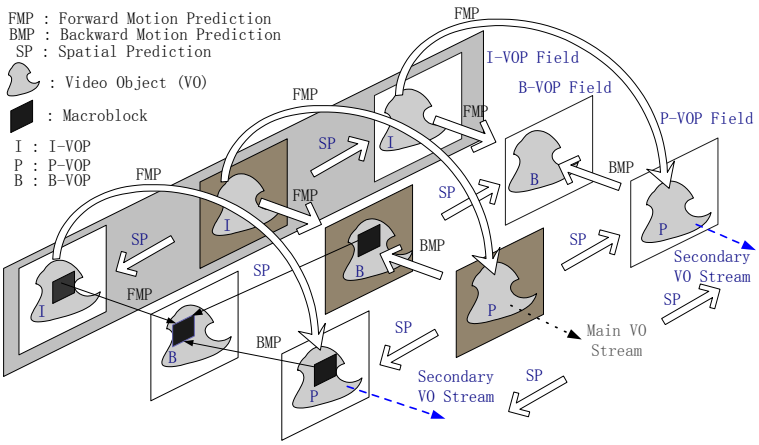

Fig. 4. Proposed method for the texture coding of an IBR object in the plenoptic video.

Fig. 5. Typical rendering results for the PV "synthesis" and "dancer" (Top); IBR object "ball" and "dancer" (Bottom).

Fig. 6. Configuration of our multiple video cameras system.

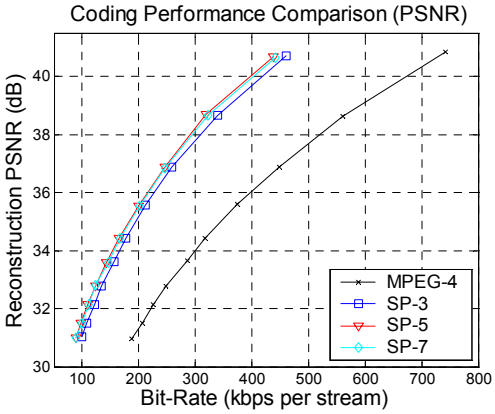

Fig. 7. Object-based coding result for the IBR object "ball".

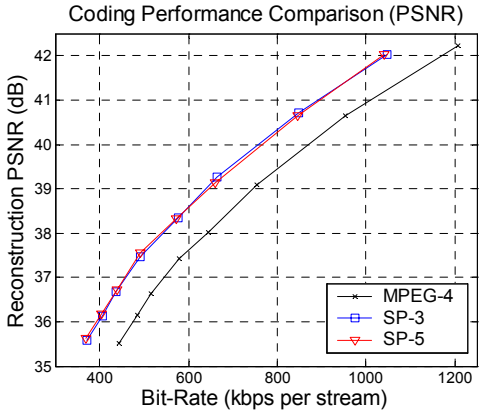

Fig. 8. Object-based coding result for the IBR object "dancer".

5. CONCLUSION

A new object-based compression scheme for a class of dynamic image-based representations called the plenoptic videos is presented. It supports functionalities at the object level and exploits both the temporal and spatial redundancy among VO streams in the videos to achieve higher compression efficiency in texture coding, shape coding, alpha map coding and depth coding. The flexibility to manipulate and render individual IBR objects and its coding performance are also demonstrated.

REFERENCES

[1] H.Y. Shum, S.B. Kang and S.C. Chan, "Survey of Image-Based Representations and Compression Techniques," in *IEEE Trans. Circuits and System for Video Technology*, vol. 13, pp. 1020-1037, Nov. 2003.

[2] S. C. Chan, K. T. Ng, Z. F. Gan, K. L. Chan and H. Y. Shum, "The plenoptic videos: capturing, rendering and compression," in *Proc. of IEEE ISCAS'04*, vol. 3, pp. 905-908, May 2004.

[3] M. G. Strintzis and S. Malasiotis, "Object-based coding of stereoscopic and 3D image sequences: A review," *IEEE Signal Processing Mag.*, vol. 16, pp. 14–28, May 1999.

[4] M. E. Lukacs, "Predictive coding of multi-viewpoint image sets," in *Proc. of IEEE ICASSP'86*, pp. 521–524, 1986.

[5] J. X. Chai, X. Tong, S.C. Chan and H.Y. Shum, "Plenoptic sampling," in *Proc. of SIGGRAPH'00*, pp. 307–318, July 2000.

[6] H. Y. Shum, J. Sun, S. Yamazaki, Y. Li and C. K. Tang, "Pop-up light field: An interactive image-based modeling and rendering system," *ACM Trans. on Graphics*, vol. 23, issue 2, pp. 143 -162, April 2004.

[7] S. C. Chan, K. T. Ng, Z. F. Gan, K. L. Chan and H.Y. Shum, "The Compression of Simplified Dynamic Light Fields," in *Proc. of IEEE ICASSP'03*, vol. 3, pp. 653-656, Hong Kong, Apr. 2003.

[8] ITU-T Recommendation ISO/IEC 14496-2:2001, "Information Technology- Coding of audio-visual objects -- Part 2: Visual".

[9] E. H. Adelson and J. Bergen, "The plenoptic function and the elements of early vision," in *Computational Models of Visual Processing*, pp. 3-20, MIT Press, Cambridge, MA, 1991.

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

[11] S. J. Gortler, R. Grzeszczuk, R. Szeliski and M. F. Cohen, "The lumigraph," in *Proc.of SIGGRAPH'96*, pp. 43-54, Aug. 1996.

[12] Y. Li, J. Sun, C. K. Tang and H. Y. Shum, "Lazy snapping," in *Proc. of SIGGRAPH'04*, pp.303-308, 2004.

[13] F. Bossen and T. Ebrahimi, "A simple and efficient binary shape coding technique based on bitmap representation," in *Proc. of IEEE ICASSP'97*, vol. 4, pp. 3129-3132, Munich, Germany, Apr., 1997.

[14] MPEG-4 video verification model v18.0, ISO/IECJTC1/SC19/ WG11 Coding of Moving Pictures and Audio N3908, Pisa, Jan. 2001.