AN EFFICIENT MULTIVIEW VIDEO COMPRESSION SCHEME

Baochun Bai, Pierre Boulanger, and Janelle Harms

Department of Computing Science, University of Alberta, Edmonton, AB, Canada T6G 2E8
{baochun, pierreb, harms}@cs.ualberta.ca

ABSTRACT

Multiview video compression is important to the image-based 3D video applications. In this paper, we proposes a novel neighbor-based multiview video compression scheme. It is essentially a MPEG2-like block-based scheme. In particular, a method to decide the stream encoding order is presented. The resulting stream encoding order can better decorrelate spatial redundancies among multiple video streams than the center approach. Experimental results confirm the superiority of the proposed neighbor approach over the center approach and MPEG2 for multiview video compression.

1. INTRODUCTION

With the advancement of computer graphics and vision technologies, 3D video [1], [2], [3], [4], [5], [6], [7], [8], [9] will become a reality in the near future. Multiview video compression is a critical part to the success of a real-time 3D video application over networks. 3D video can offer the viewers arbitrary viewpoints to the dynamic scene and thus allow them to enjoy a feeling of immersion into an event such as an Olympic competition or a popular theater show. It is inherently different from traditional video such as those shown on today's TV. Traditional video is essentially a two-dimensional medium and only provides a passive way for viewers to observe the event. 3D video allows viewers to actively move their own eyes to interested targets that are not the focus of the cameraman. 3D video normally needs to capture multiple synchronized video streams, which is usually called multiview video, and new viewpoints are created by advanced graphics rendering algorithms such as Image-Based Rendering (IBR) [10], [11].

3D video can be created by two general approaches. One is a model-based approach, which acquires multiview video from sparsely-arranged cameras and extracts 3D models from the images [1], [3], [4], [5], [12], and then renders the new viewpoints with the help of a 3D scene model. The model-based approach has the advantage of reducing the acquisition cost by using few cameras. However, it increases the algorithmic complexity. In addition, it is still a very difficult problem to extract scene models from general real-world scenes in real time. The other approach is a pure image-based approach, which uses densely arranged cameras to acquire high-resolution light fields and then uses image-based rendering [13] to generate images at the new viewpoints [2], [14], [15]. The image-based approach has the advantage of reconstructing new views without the scene model. It is our belief that the image-based approach will be the solution to 3D video in the near future though the model-based approach certainly will become a major approach to 3D video should it be available one day. However, an image-based approach demands more storage and transmission bandwidth for multiview video data. It makes multiview video compression indispensable for the practical use of 3D video. In this paper, we will propose an efficient block-based multiview video compression algorithm.

Multiview video compression needs to simultaneously reduce temporal and spatial redundancy among multiple synchronized video streams. MPEG-2 Multiview Profile (MVP) [16] proposes a block-based stereoscopic coding (BBSC) to encode the stereo video. Motion-compensated prediction (MCP) is used to reduce the temporal redundancies and disparity-compensated prediction (DCP) is used to reduce the spatial redundancies. MVP first compresses, say, the left view, with a MCP-based monoview coding algorithm and then encodes the right view using both DCP from the left view and MCP. [17] uses a mesh-based disparity estimation method to improve the decorrelation of spatial coherence. [18], [19] extends MVP to compress multiview video sequences. Multiple video streams are classified into two types of streams: main stream and secondary stream. The main stream is the central stream among all the streams and is encoded using only a MCP-based MPEG2-like algorithm. Secondary streams are compressed with MCP and DCP based on the main stream as illustrated in Fig. 1(b). We call this approach the *center approach*.

In this paper, we put forward an efficient MPEG2-like blocked-based multiview video compression scheme, which also classifies video streams into main stream and secondary stream. One stream is chosen as the main stream and is coded using only a MCP-based MPEG2-like algorithm. Unlike the center approach, the secondary stream is compressed using DCP from multiple nearest neighbors and MCP. We call our approach *the neighbor approach*. Our approach can have better spatial redundancy reduction than the center approach and thus improve the video quality under the same bit rate.

The rest of the paper is organized as follows. Section 2 discusses in detail the proposed multiview video compression scheme. In Section 3, we present the experimental results. The paper concludes in Section 4.

2. NEIGHBOR-BASED MULTIVIEW VIDEO COMPRESSION

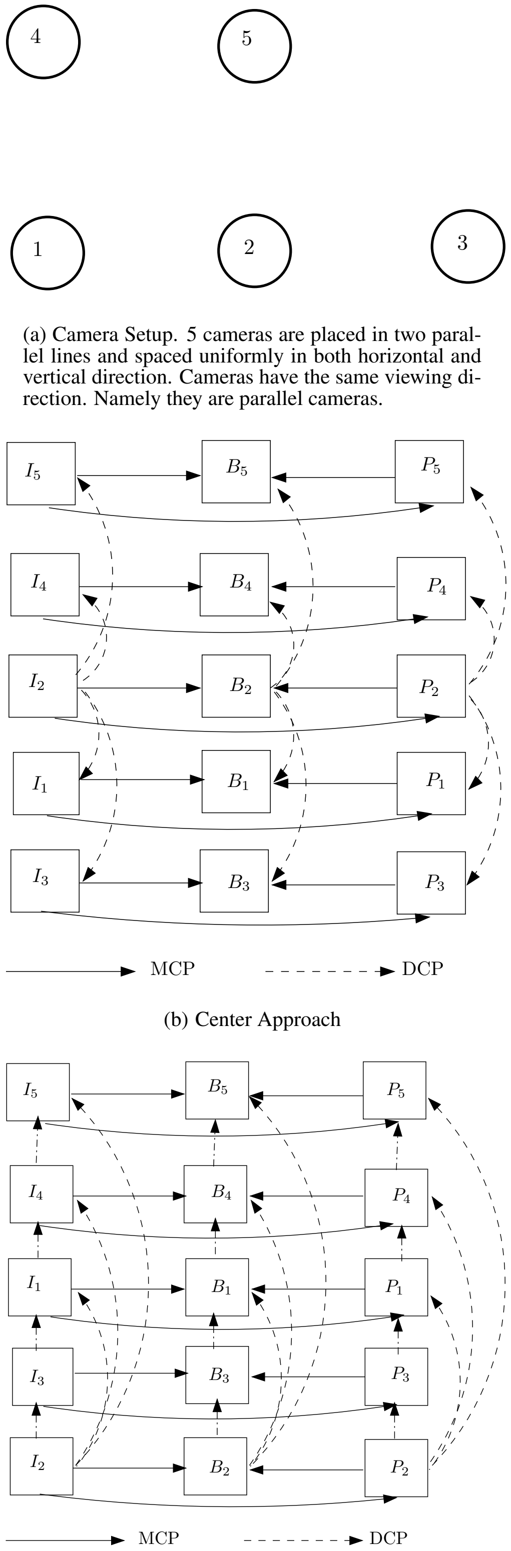
The proposed neighbor-based multiview video compression is a MPEG2-like block coder. Namely, a macroblock is used as a basic unit for motion and disparity estimation and prediction. Similar to the center approach, each stream is one of two types: main stream and secondary stream. There is only one main stream. The pictures are also classified into three types of pictures: I, P, and B-frame. I, P, and

B pictures in the main stream are encoded with a MPEG-2 like algorithm. However, pictures in the secondary streams are encoded with a different method. I pictures in the secondary streams are encoded with DCP based on the I pictures in neighbor streams. P and B pictures are encoded with MPEG2-like MCP and DCP of corresponding P and B pictures in neighbor streams. The scheme is illustrated in Fig. 1(c). Each stream uses the same group of picture (GOP) structure as MPEG2. All the synchronized GOPs of multiview video form a group of GOP (GGOP) similar to the center approach. In this paper, we assume that each camera position and its viewing direction are known.

The basic idea of the neighbor approach is to use multiple nearest neighbor frames for disparity-compensated prediction. However, the synchronized frames in multiview video are coded in a specific order and the early-coded frames may not find optimal nearest neighbor frames as reference frames. For example, in Fig. 1(a), suppose that a frame can have at most two neighbor frames as reference frames for DCP. The stream encoding order is $S_2S_3S_1S_4S_5$, where S_i denotes stream i. The frames in S_4 can only have frames in stream S_1 and S_2 as reference frames though obviously its two nearest neighbor streams are S_1 and S_5. The key challenge of the neighbor approach is to decide the stream encoding order so that the number of streams which have the optimal neighbor streams as reference frames can be maximized and thus the maximal decorrelation of spatial coherence can be achieved. We use Algorithm 1 to decide the stream encoding order.

The algorithm first finds m nearest neighbor streams for each stream. For parallel camera setup, Euclidean distance between two cameras can be used as a metric to measure the closeness between two cameras. For convergent camera setup, the angle between the viewing directions of two cameras can be used as a metric to measure the closeness between two cameras. The $count_i$ is increased by one if the stream i is deemed as one of the closest neighbors by another stream. The stream with the largest $count_i$ is chosen as the main stream. The main stream is the "central" stream in the sense that it is the one that is deemed as the nearest neighbors by most other streams. The main stream is added to the coded stream set. Then the stream in the neighbor set of the main stream is added to the set cur of streams that are potentially coded next. The stream with the largest count in the set cur is chosen to be coded next. The rationale is that the next coded stream should have the maximal spatial redundancies with coded frames while at the same time it should make the later-coded streams have a higher probability to find optimal neighbors. The algorithm terminates when all the streams are coded. Namely the stream encoding order is decided.

We currently use the rate control algorithm of the MPEG2 Test Model 5 [20] to prevent buffer overflow and underflow problems. The main stream is allocated a higher bit rate than the secondary stream since the main stream is only motion-compensated and we intend to maintain a uniform reconstruction quality among all the streams. More complicated bit rate control algorithm such as joint coding of multiview video [21] is now under investigation. Like MPEG2, DCT-based transform is used to code the prediction error and the coefficients are then entropy coded using Huffman or arithmetic coding. The algorithm needs $2n + k$ buffers to do motion and disparity prediction, where $m \leq k < n$ and n is the number of video streams and m is the maximal number



(a) Camera Setup. 5 cameras are placed in two parallel lines and spaced uniformly in both horizontal and vertical direction. Cameras have the same viewing direction. Namely they are parallel cameras.

(b) Center Approach

(c) Neighbor Approach. Each frame can have maximal two neighbor frames as reference frames.

Fig. 1. Illustration of the Center Approach and the Neighbor Approach

Algorithm 1 Pseudo Code to Decide The Stream Encoding Order

```
$n$ : the total number of streams;
$m$ : the maximal allowed neighbor frames for each frame;
$S_i$ : stream $i$;
$sall$ : the set of all the $n$ streams;
$nbr_i^m$ : the set of $m$ neighbors of stream $i$;
$count_i$ : the number of times for stream $S_i$ in $nbr_j^n$, $i \neq j$;
$coded$ : the set of coded streams;
$cur$ : the set of streams to be coded;
$i$, $j$, $k$ $u$: integer variables;

for $i$ = 1 to $n$ do
    compute $nbr_i^m$ for $S_i$;
    increase $count_i$ by 1 if $S_i$ is in $nbr_j^m$;
end for
$S_k$ is chosen as the main stream, where $count_k \geq count_i$, $i \neq k$ and $i \in \{1 \dots n\}$;
$coded = coded \cup S_k$;
output $S_k$;
$cur = cur \cup nbr_k^m - coded$;
while $coded \neq sall$ do
    if $cur \neq NULL$ then
        choose the stream $S_u$ with the largest $count_u$ in $cur$ as the next coded stream;
        $cur = cur \cup nbr_u^m - S_u - coded$;
    else
        choose the stream $S_u$ with the largest $count_u$ in $sall - coded$ as the next coded stream;
        $cur = cur \cup nbr_u^m - coded$;
    end if
    update $nbr_u^m$ so that its $m$ nearest neighbor streams are in the set $coded$.
    $coded = coded \cup S_u$;
    output $S_u$;
end while
```

of neighbor streams.

3. EXPERIMENTAL RESULTS

Synthetic video is used to evaluate the proposed multiview compression scheme due to the difficulty to acquire synchronized real-world multiview video. The video is rendered using POVRAY [22]. Cameras are placed in parallel lines and spaced uniformly in both the horizontal and vertical direction. This is a parallel camera setup and the viewing direction is the same for every camera. Fig.2 shows part of the synthesized multiview video used in this research. Each stream is a $24 - bit$ RGB video with 640×480 pixels and 25 frames per second. Fig. 3 gives a video quality comparison among the neighbor approach, the center approach and MPEG2 for the case of 14 streams with 7 cameras per line. The PSNR is the averaged PSNR for 14 streams. It shows that the neighbor approach has a better video quality than both the center approach and MPEG2. This is expected since the neighbor approach uses the nearest neighbor frames as reference frames and can better exploit the spatial coherence than the center approach and MPEG2 that uses only temporal prediction and no spatial prediction. In addition, the figure shows that as the number of the neighbor frames increases, the reconstructed video quality also increases since more neighbor frames lead to more spatial redundancy reduction. However, PSNR improvement is limited as the number of neighbor frames increases. This is understandable since only a limited number of macroblocks will be predicted by new added neighbor frames while most other macroblocks already have a small prediction error and it is hard to further diminish the prediction error with the added neighbor frames. We also conducted many other experiments by varying the number of streams and the bit rate. The results are similar and we omit them for brevity.

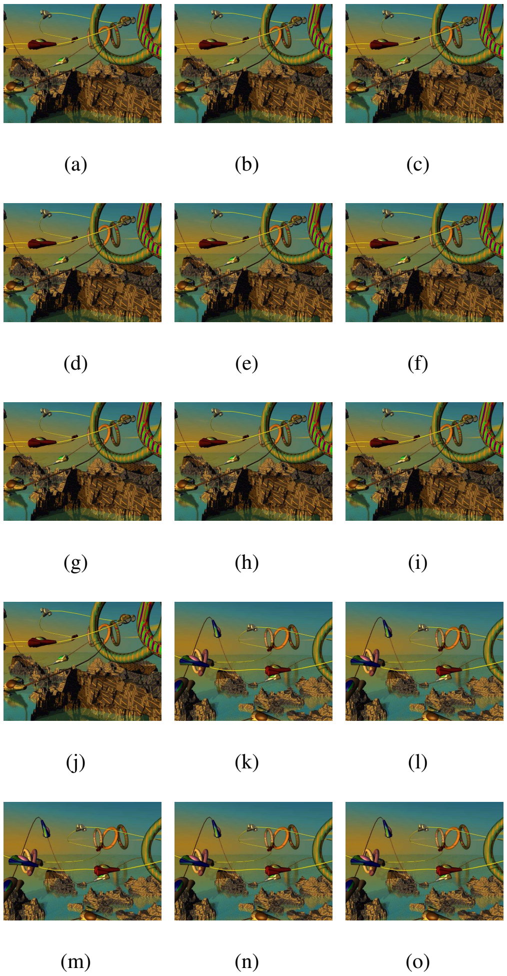


Fig. 2. Multiview Video

Each macroblock in the proposed compression scheme can be predicted using one of 6 modes: Forward (F), Backward (B), Disparity (D), Forward and Backward interpolation (FB), Forward and Disparity Interpolation (FD), Backward and Disparity Interpolation (BD). Table I shows the average percentage of macroblocks used for disparity prediction related types in both the center and the neighbor approach at the bit rate 1Mbps for each stream in the case of 14 streams. It indicates that disparity-related compensation accommodates most prediction types and plays a significant role to reduce prediction error and improve the image quality. The results also show that the neighbor approach has a higher percentage macroblocks predicted by dispar-

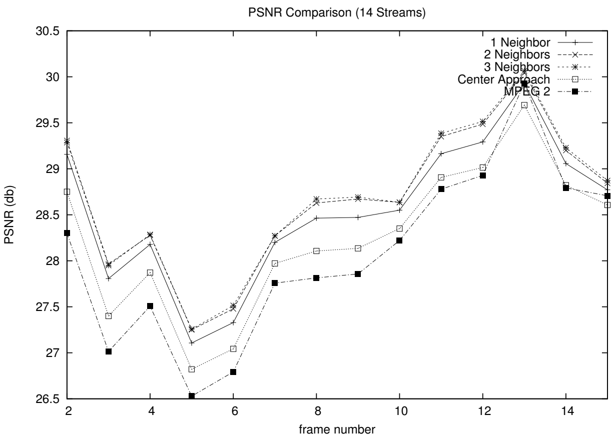


Fig. 3. PSNR Comparison (14 Steams)

ity than the center approach. Thus the better image quality in the neighbor approach should obviously be attributed to the spatial redundancy reduction. From the data, it is also evident that the benefits from increasing the number of neighbor frames is limited and the gain becomes smaller and smaller as the the number of neighbor frames continually increases, which is clearly shown in Fig 3.

Table I

THE PERCENTAGE OF MACROBLOCKS USED FOR DISPARITY PREDICTION RELATED MODES (14 STREAMS)

Modes	D	FD	BD	SUM
Center	34.3%	11.6%	20.1%	66%
1 Neighbor	44.3%	10.4%	18.2%	72.9%
2 Neighbors	45.8%	10.3%	20.8%	76.4%
3 Neighbors	48.9%	9.4%	18.9%	77.2%

4. CONCLUSIONS

In this paper, we present a novel neighbor-based multiview video compression scheme. It is a MPEG2-like block-based video compression scheme. In particular, we put forward an efficient algorithm to find a nearly optimal stream encoding order which maximizes spatial redundancy reduction. We compared the proposed scheme with the center approach and MPEG2. Experimental results show that the proposed compression scheme can achieve better video quality over the center approach and MPEG2 under the same bit rate.

5. ACKNOWLEDGEMENT

The first author is grateful to the Province of Alberta Graduate Fellowship to make this research possible. We are also thankful to the grant support from NSERC (Natural Science and Engineering Research Council of Canada).

6. REFERENCES

[1] C. Fehn, E. Cookie, O. Schreer, and P. Kauff, “3d analysis and image-based rendering for immersive tv applications,” *Signal Processing: Image Communications*, vol. 17, pp. 705–715, 2002.

[2] W. Matusik and H. Pfister, “3d tv: A scalable system for real-time acquisition, transmission and autostereoscopic display for dynamic scenes,” in *Proc. of ACM SIGGRAPH, 2004*, August 2004.

[3] C. L. Zitnick, S. B. King, M. Uyttendaele, S. Winder, and R. Szeliski, “High-quality video view interpolation using a layered representation,” in *Proc. of ACM SIGGRAPH, 2004*, August 2004.

[4] M. Gross, S. Wurmlin, M. Naef, E. Lamboray, C. Spagno, A. Kunz, E. Koller-Meier, T. Svoboda, L. V. Gool, S. Lang, K. Strehlke, A. V. Moere, and O. Staadt, “blue-c: A spatially immersive display and 3d video portal for telepresence,” *ACM Transactions on Graphics*, vol. 22, no. 3, pp. 819–828, 2003.

[5] J. Carranza, C. Theobalt, M. A. Magnor, and H. P. Seidel, “Free-viewpoint video of human actors,” *ACM Transactions on Graphics*, vol. 22, no. 3, pp. 569–577, 2003.

[6] A. Smolic and P. Kauff, “Interactive 3d video representation and coding technologies,” *Proceedings of the IEEE, Special Issue on Advances in Video Coding and Delivery*, vol. 93, no. 1, pp. 98–110, January 2005.

[7] M. Magnor, “3d-tv - the future of visual entertainment,” in *Proc. of Multimedia Databases and Image Communications (MDIC’04)*, June 2004.

[8] A. Smolic, C. Fehn, and K. Mueller, “Mpeg 3dav - video-based rendering for interactive tv applications,” in *Proc. 10. Dortmunder Fernsehseminar, ITG/FKTG-Fachtagung*, September 2003.

[9] A. Smolic, K. Mueller, P. Merkle, T. Rein, P. Eisert, and T. Wiegand, “Free viewpoint video extraction, representation, coding, and rendering,” in *Proc. of IEEE International Conference on Image Processing (ICIP 2004)*, October 2004.

[10] H.-Y. Shum, S. B. Kang, and S.-C. Chan, “Survey of image-based representations and compression technique,” *IEEE Transaction on Circuits and Systems for Video Technology*, vol. 13, no. 11, pp. 1020–1037, November 2003.

[11] C. Zhang and T. Chen, “A survey on image-based rendering–representation, sampling and compression,” *Signal Processing: Image Communications*, vol. 19, no. 1, pp. 1–28, 2004.

[12] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen, “The lumigraph,” in *Proc. of ACM SIGGRAPH 1996*, 1996.

[13] M. Levoy and P. Hanrahan, “Light field rendering,” in *Proc. of ACM SIGGRAPH 1996*, 1996.

[14] J. C. Yang, M. Everett, C. Buehler, and L. McMillan, “A real-time distributed light field camera,” in *Proc. of the 13th Eurograhics Workshop on Rendering*, 2002.

[15] T. Fujii and M. Tanimoto, “Free-viewpoint tv system based on ray-space representation,” in *Proc. of SPIE ITCom, Vol. 4864-22*, 2002.

[16] ISO/IEC 13818-2 AMD 3, “Mpeg-2 multiview profile,” *ISO/IEC JTC1/SC29/WG11 document no. N1366*, September 1996.

[17] R.-S. Wang and Y. Wang, “Multiview video sequence analysis, compression, and virtual viewpoint synthesis,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 10, no. 3, pp. 397–410, April 2000.

[18] S.-C. Chan, K.-T. Ng, Z.-F. Gan, K.-L. Chan, and H.-Y. Shum, “The compression of simplified dynamic light field,” in *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2003)*, April 2003.

[19] J. Lim, K. N. Ngan, W. Yang, and K. Sohn, “A multiview sequence codec with view scalability,” *Signal Processing: Image Communication*, vol. 19, no. 3, pp. 239–256, March 2004.

[20] ISO/IEC-JTC1/SC29/WG11 MPEG 93/457, “Test model 5,” April 1993.

[21] L. Wang and A. Vincent, “Bit allocation and constraints for joint coding of multiple video programs,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 6, pp. 949–959, Sept. 1999.

[22] “http://www.povray.org,” .