ADAPTIVE APPEARANCE COMPENSATED VIEW SYNTHESIS PREDICTION FOR MULTIVIEW VIDEO CODING

Shinya Shimizu, Hideaki Kimata, and Yoshimitsu Ohtani

NTT Cyber Space Laboratories, NTT Corporation

ABSTRACT

View synthesis prediction has been studied to achieve efficient inter-view prediction. Existing view synthesis prediction methods synthesize predicted pictures by using decoded pictures from the other views and geometric information of the scene. However, these conventional methods have no ability to compensate the inter-view difference in image signals caused by individual camera characteristics and the non-Lambert reflection of objects.

This paper proposes adaptive appearance compensated view synthesis prediction; it uses adaptive filters to compensate not only the differences in color and brightness but also the differences in the spatial frequency of images. The object-dependent local differences are also compensated efficiently by estimating the optimal compensation parameters block by block at the decoder. Experiments show that the proposed method reduces the bitrate by up to 40% relative to H.264/AVC Multiview Video Coding, and 26% relative to the conventional view synthesis prediction.

Index Terms— Multiview video coding, View synthesis prediction, Adaptive filter, Depth map

1. INTRODUCTION

Advanced visual media have been explored for many years. Free-viewpoint TV (FTV) and three-dimensional video (3D Video) are attracting a lot of interest as such services [1, 2]. Multiview video is one of the rawest representations for these advanced three-dimensional visual media. Recent progress on technologies for multiview video processing will make such services possible in the near future.

Although MPEG-4 AVC/H.264 Annex.H Multiview Video Coding (MVC) achieves efficient encoding for multiview video, the bitrate yielded by MVC is proportional to the number of views. Since a large number of views are required for FTV and multiview auto-stereoscopic displays, it is desirable to be able to generate videos at arbitrary viewpoints from just a limited number of views. View generation can be achieved by using depth information of the scene [3]. In order to support low complexity view generation at the display side, a lot of research is targeting the transmission of both multiview video and multiview depth maps [4, 5]. One disadvantage of this approach is the bitrate increase needed to transmit the depth information.

View synthesis prediction (VSP) is one of the promising technologies with which multiview video can be encoded efficiently by using depth information [6]. A predicted picture is synthesized by warping the image signals of the reference pictures into the coding target view. Compared to disparity compensated prediction, VSP can finely compensate scene geometry. However, current implementations fail to compensate the inter-view mismatch caused by individual camera characteristics and non-Lambert reflection of objects. As a result, existing VSP techniques fail to reduce the bitrate drastically.

In this paper, we propose adaptive appearance compensated view synthesis prediction; it uses block adaptive filters to reduce the inter-view mismatch of image signals. In Section 2, we describe the conventional VSP, and the proposed method is presented in Section3. Section 4 introduces the experimental results, and Section 5 concludes this paper.

2. VIEW SYNTHESIS PREDICTION (VSP)

VSP offers efficient inter-view prediction for multiview video coding. VSP increases the prediction accuracy by using scene geometry and camera parameters. VSP generates the predicted picture by using pixel correspondences. The corresponding pixels are identified by the inverse-projection of pixels and the re-projection of reconstructed three-dimensional points. Eq. 1 defines the inverse-projection; object position g in the 3D world is reconstructed from pixel position (u_a, v_a) and camera-object distance d. The re-projection is given by Eq. 2; pixel position (u_b, v_b) in other view is obtained by following the process of shooting.

$$g = R_a^{-1} A_a^{-1} \left(u_a, v_a, 1\right)^T d - t_a \tag{1}$$

$$k \left(u_b, v_b, 1\right)^T = A_b R_b \left(g + t_b\right) \tag{2}$$

, where A, R, and t denote the intrinsic matrix, rotation matrix, and translation vector of the camera, respectively. k is a scalar value and subscripts, a and b, denote views. Some previous works take a as the reference view [7] and others take

Thanks to the Interactive Visual Media Group at Microsoft Research for providing *breakdancers* sequence.

b as the reference view [6], but there is no inherent difference in the quality of synthesized views. Our VSP takes b as the reference view.

VSP compensates disparities pixel by pixel, so it can achieve geometrically more precise prediction than block-based disparity compensated prediction. However, VSP has no ability to predict inter-view inconsistencies in the image signals because VSP uses image signals from the other views as predicted signals.

Many studies have tried to overcome this problem. Yamamoto it et al. proposed color compensation via look-up-tables (LUT) [8]. This method can handle inter-view illumination changes, but not inter-view focus mismatch. Moreover, the LUTs are so large that it is difficult to achieve efficient compression. Adaptive reference filtering (ARF) was proposed to compensate the inter-view focus mismatch [9]. ARF utilizes 2D filters to generate additional inter-view reference pictures whose focus is similar to that of the coding target picture. ARF uses multiple filters at one frame to compensate object-dependent mismatch and all filter coefficients are encoded. As a result, ARF fails to achieve efficient coding with the scene where there are a lot of objects.

In order to compensate object-dependent inter-view color/focus mismatch efficiently, this paper proposes adaptive appearance compensated view synthesis prediction. The proposed method estimates the filter coefficients block by block at the decoder side while ARF decides and encode them at the encoder side.

3. ADAPTIVE APPEARANCE COMPENSATED VIEW SYNTHESIS PREDICTION (AAC-VSP)

3.1. Adaptive Filter for Appearance Compensation

Inter-view differences of image signals are mainly caused by the different camera-object distances, camera heterogeneity, and the non-Lambert reflection of objects. Such inter-view differences fall into two categories.

One is the differences in color and brightness. This kind of inter-view difference is usually compensated by using a LUT or mapping function. However, it is difficult to estimate a global model that can compensate such differences because mapping may be non-linear and position-dependent. The other is the differences in image spatial frequency. A lens is in precise focus at just one camera-object distance and the sharpness gradually falls away from this distance. The differences may be invisible within the depth of field, but they surely exist and may preclude efficient encoding in the frequency domain.

The proposed method uses adaptive filters with offset, see Eq. 3, to compensate these inter-view differences efficiently.

$$Comp[x,y] = \sum_i \sum_j F_{i,j} Syn[x+i, y+j] + o \quad (3)$$

, where $Comp$ denotes view synthesis image signals after compensation and Syn denotes the view synthesis image signals generated by warping the image signals on reference views. $F_{i,j}$ and o denote filter coefficient and offset, respectively. This paper refers to a set of filter coefficients and offset as the compensation parameters.

The degree of inter-view mismatch varies with not only sequences, but also time. For efficient compensation, the proposed method provides three compensation granularities. One granularity is selected slice by slice given an assessment of coding performance. The lowest level is no compensation; the warped image signals are used in VSP. Next level allows the use of just one set of compensation parameter in one slice. The last level provides block-adaptive compensation; the optimal parameters are estimated block by block. The last level can compensate non-linear, position-dependent, and/or object-dependent differences.

When using slice level compensation, the encoder calculates the optimal parameter by minimizing the energy of prediction error and encodes it as in [9]. In the case of block level compensation, we propose to derive compensation parameters at the decoder because the number of parameters is too large to achieve efficient compression. The derivation process is described in the next subsection.

3.2. Compensation Parameter Derivation in Decoder

The encoder can estimate the optimal compensation parameters by minimizing the differences between the view synthesized picture and the coding target picture because the coding target pictures can be obtained. However, decoder has no way to get the coding target picture itself.

The purpose of video coding is preserving the image signals within the bitrate limit. Therefore, it is possible to assume that the decoded pictures are almost the same as the original ones. Based on this hypothesis, the proposed method uses already decoded image signals around the target block to estimate the optimal compensation parameters.

For block level compensation, the proposed method derives the optimal parameters by applying the least-square method to the following estimated prediction error on neighboring pixels. This derivation is performed block by block in both encoder and decoder.

$$Err = \sum_{(h,w)\in Ref} (Comp[h,w] - Dec[h,w])^2 \quad (4)$$

, where Dec denotes the decoded picture and Ref is a set of already coded neighboring pixels as illustrated in Fig. 1.

4. EXPERIMENT

4.1. Conditions

We implemented the proposed method on JMVM version 8.0 [10], which is the test model of H.264/AVC Annex.H Multi-

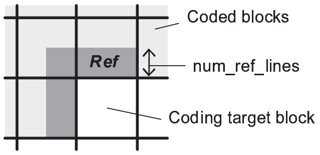

Fig. 1. Reference pixels for the derivation of parameters

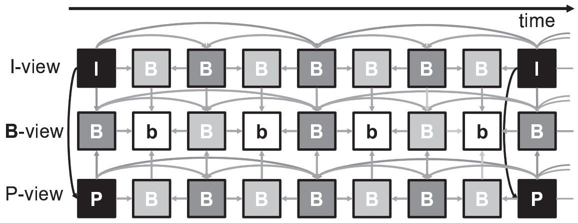

Fig. 2. Hierarchical B structure for MVC(GOP Size 8)

view video coding (MVC), and conducted an experiment to assess the efficiency of our method.

VSP was implemented as one of the Inter16x16 modes. This means only 16x16 pixel units were allowed. To distinguish VSP mode from the normal Inter16x16 modes, one bit flag was encoded. The VSP macroblock did not encode the reference index, motion/disparity vector, transform flag, or prediction residuals, like the skip macroblock.

We used a 5x5 filter for the luminance component and a 3x3 filter for the chrominance components, so the total number of coefficients was 46. Therefore, forty-six 32-bit float point numbers were encoded in the slice header when slice level compensation was selected in the process of the optimization. Block level compensation parameters were derived for each macroblock. num_ref_lines was set to 8 in the experiments.

We used three MVC test sequences; *akko&kayo*, *breakdancers*, and it rena. Depth maps for *breakdancers* were provided by Microsoft Research [4], and depth maps for the others were estimated from multiview video by using the graph-cut stereo algorithm, one of the most popular stereo algorithms. All the multiview depth maps were encoded by MVC with the Basis QP equal to 36.

We encoded three views on each sequence with the reference structure illustrated in Fig. 2. A hierarchical B structure (GOP 8) was applied in the temporal direction. There were one I-view, an H.264/AVC compatible view, one P-view, where inter-view prediction is allowed in one direction, and one B-view, where inter-view prediction is allowed in both directions. The other important coding settings are listed in Table 1.

4.2. Experimental Results

Fig. 3 plots the rate-distortion curves for *breakdancers*, *akko&kayo*, and *rena*. The ”MVC” curves plot the coding results for H.264/AVC Annex.H MVC. The ”VSP” curves show the coding performance for VSP with no compensation. The ”AAC-VSP” curves are for the proposed method.

As can be seen, the proposed method achieves the best performance at any rate for all sequences. Unlike conventional VSP, the proposed method is very effective at high bitrates. This is because the proposed method succeeds in increasing the prediction accuracy. However, the proposed method offers only limited improvements at low bitrates. This might be because low bitrate decoded pictures include a lot of coding noise. In such cases, our hypothesis that the decoded picture is almost the same as the original picture is false, so the codec fails to estimate the optimal compensation parameters. This consideration is supported by the fact that at low bitrates the proposed method set a lot of slices to the no compensation mode.

Table 2 shows the bitrate reductions and PSNR gains relative to MVC or VSP in the Bjøntegaard measure [11]. As can be seen, the proposed method shows remarkable performance. Since the degree of inter-view mismatch depends on the sequence, the improvements also depend on the sequence. The best case, for the *breakdancers* sequence, is the bitrate reduction of more than 40% relative to the MVC standard and about 27% relative to the conventional VSP scheme. The bitrate reduction on the other sequences ranges from about 11% against MVC and about 7% against the conventional VSP on average. These correspond to PSNR improvements of about 0.5dB and 0.3dB, respectively.

5. CONCLUSION

We proposed a new view synthesis prediction method named adaptive appearance compensated view synthesis prediction. The proposed method can compensate not only the differences in color and brightness but also the spatial frequency of images by using adaptive filters. The local differences are also compensated efficiently by estimating the compensation parameters block by block at the decoder side. An experiment shows that the proposed method reduces the bitrate by up to

Table 1. Coding parameters

Parameter	Value
GOP size	8
Anchor period	8
Number of reference frames	2
Motion estimation scheme	FME
Entropy coding method	CABAC
Hadamard transform	used
RD-optimized mode decision	used
Layer0 QP (Basis QP)	22, 27, 32, 37
Layer1 QP	Layer0 QP + 3
Layer2 QP	Layer1 QP + 1
Layer3 QP	Layer2 QP + 1

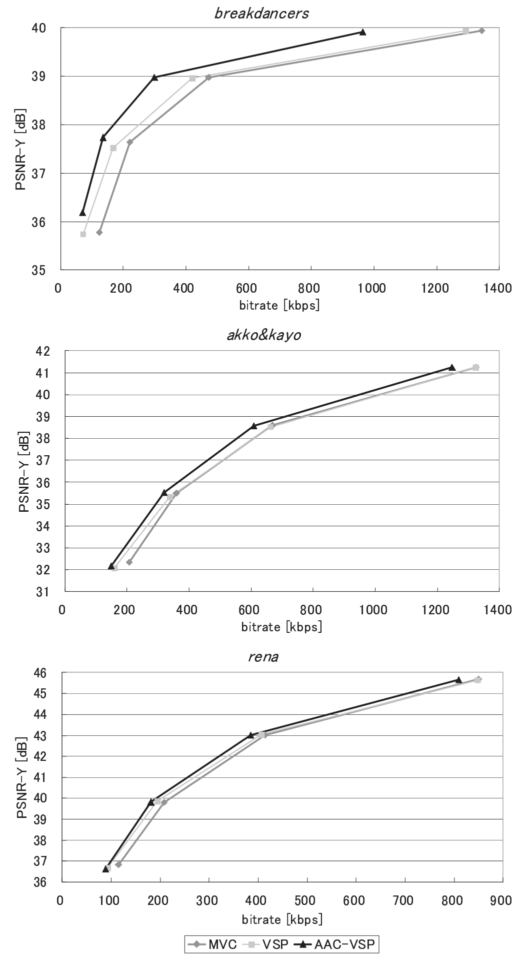

Fig. 3. Overall RD performance

40% and about 20% on average for 3 sequences relative to H.264/AVC Annex.H Multiview Video Coding.

In this paper, multiview depth maps were encoded at the same bitrate even if the total target bitrate was changed. It is obvious that the accuracy and quality of the depth maps affects the coding performance. Therefore, one future work is studying the effects of depth maps. Furthermore, we plan to expand our VSP scheme with encoding prediction residuals and consider smaller block sizes for both the compensation parameter derivation and VSP processes.

6. REFERENCES

[1] Masayuki Tanimoto, “Overview of free viewpoint television,” *Signal Processing: Image Communication*, vol. 21, no. 6, pp. 454–461, July 2006.

[2] Aljoscha Smolic, Karsten Müller, Philipp Merkle, Christoph Fehn, Peter Kauff, Peter Eisert, and Thomas Wiegand, “3d video and free viewpoint video - technologies, applications and mpeg standards,” in *Proc. ICME2006*, July 2006, pp. 2161–2164.

[3] Jin-Xiang Chai, Xin Tong, Shing-Chow Chan, and Heung-Yeung Shum, “Plenoptic sampling,” in *Proc. ACM SIGGRAPH 2000*, 2000, pp. 307–318.

[4] C. Lawrence Zitnick, Sing Bing Kang, Matthew Uyttendaele, Simon Winder, and Richard Szeliski, “High-quality video view interpolation using a layered representation,” *ACM Trans. Graph.*, vol. 23, no. 3, pp. 600–608, 2004.

[5] Aljoscha Smolic, Karsten Müller, Kristina Dix, Philipp Merkle, Peter Kauff, and Thomas Wiegand, “Intermediate view interpolation based on multiview video plus depth for advanced 3d video system,” in *Proc. ICIP2008*, 2008, pp. 2448–2451.

[6] Sehoon Yea and Anthony Vetro, “View synthesis prediction for rate-overhead reduction in ftv,” in *Proc. 3DTV-Conference*, May 2008, pp. 145–148.

[7] Yuichi Taguchi and Takeshi Naemura, “View-dependent coding of light fields based on free-viewpoint image synthesis,” in *Proc. ICIP2006*, October 2006, pp. 509–512.

[8] Kenji Yamamoto, Masaki Kitahara, Hideaki Kimata, Tomohiro Yendo, Toshiaki Fujii, Masayuki Tanimoto, Shinya Shimizu, Kazuto Kamikura, and Yoshiyuki Yashima, “Multiview video coding using view interpolation and color correction,” *IEEE Trans. Circuits Syst. Video Techn.*, vol. 17, no. 11, pp. 1436–1449, 2007.

[9] Jae Hoon Kim, PoLin Lai, Joaquin Lopez, Antonio Ortega, Yeping Su, Peng Yin, and Cristina Gomila, “New coding tools for illumination and focus mismatch compensation in multiview video coding,” *IEEE Trans. Circuits Syst. Video Techn.*, vol. 17, no. 11, pp. 1519–1535, 2007.

[10] Pandit Pandit, Anthony Vetro, and Ying Chen, “Jmvm 8 software,” JVT Doc. JVT-AA208, April 2008.

[11] Gisle Bjøntegaard, “Calculation of average psnr differences between rd-curves,” VCEG Doc. VCEG-M33, April 2001.

Table 2. Simulation results (Bjøntegaard delta)

sequence	vs	BD-Rate	BD-PSNR
breakdancers	MVC	40.05%	0.75dB
	VSP	26.59%	0.45dB
akko&kayo	MVC	11.79%	0.53dB
	VSP	8.68%	0.40dB
rena	MVC	10.88%	0.46dB
	VSP	5.19%	0.22dB