MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

Backward View Synthesis Prediction for 3D-HEVC

Tian, D.; Zou, F.; Vetro, A.

TR2013-086 September 2013

Abstract

View synthesis prediction provides an effective way to reduce inter-view redundancy of multi-view video in addition to conventional disparity compensated prediction. Traditional forward warping techniques incur high complexity since an entire picture is typically warped from one viewpoint to another. To reduce this complexity, block-based backward warping is considered as an alternative solution. One difficulty with this approach is that it requires depth information of the current block prior to its encoding. To solve this problem, a novel method is proposed to derive the depth information from neighboring blocks with high accuracy. With the proposed approach, backward warping is enabled using depth information derived from either neighboring spatial (inter-view) compensated blocks or temporal compensated blocks. As a generalization, the proposed backward warping scheme is not only applied at the pixel level, but at the sub-block level as well. Simulation results demonstrate that the proposed scheme achieves an average bitrate savings of 1.2% for coded video vs coded video bitrate, 1.1% for coded video vs total bitrate, and 1.0% for synthesized video vs total bitrate under common test conditions for 3D video coding using HEVC, with maximum gains of greater than 10% for dependent views.

IEEE International Conference on Image Processing (ICIP)

BACKWARD VIEW SYNTHESIS PREDICTION FOR 3D-HEVC

Dong Tian, Feng Zou, Anthony Vetro

Mitsubishi Electric Research Labs
201 Broadway, Cambridge, MA 02139

ABSTRACT

View synthesis prediction provides an effective way to reduce inter-view redundancy of multiview video in addition to conventional disparity compensated prediction. Traditional forward warping techniques incur high complexity since an entire picture is typically warped from one viewpoint to another. To reduce this complexity, block-based backward warping is considered as an alternative solution. One difficulty with this approach is that it requires depth information of the current block prior to its encoding. To solve this problem, a novel method is proposed to derive the depth information from neighboring blocks with high accuracy. With the proposed approach, backward warping is enabled using depth information derived from either neighboring spatial (inter-view) compensated blocks or temporal compensated blocks. As a generalization, the proposed backward warping scheme is not only applied at the pixel level, but at the sub-block level as well. Simulation results demonstrate that the proposed scheme achieves an average bitrate savings of 1.2% for coded video vs coded video bitrate, 1.1% for coded video vs total bitrate, and 1.0% for synthesized video vs total bitrate under common test conditions for 3D video coding using HEVC, with maximum gains of greater than 10% for dependent views.

Index Terms— 3D Video Coding, View Synthesis Prediction (VSP), Backward VSP

1. INTRODUCTION

Nowadays, 3D video is becoming ubiquitous in both the movie industry and consumer entertainment applications. Due in part to advances in 3D display techniques, multiple viewpoints can be rendered for a scene to improve the immersive experience of a user. Along with reduced cost of manufacturing 3D displays and potential market growth, there is increasing support from content providers and broadcasters to introduce 3D services. However, due to the dramatically increased data size, the efficient compression, storage and transmission of 3D video content is still a challenging and relevant topic.

Correspondence author: D. Tian (tian@merl.com).

The data format of 3D video coding has been evolving during the past several years. Several categories of data formats have been studied and considered. The first one, referred to as the frame-compatible format, arranges the pixels from multiple texture video into a single video before encoding. This format has the benefit of utilizing existing codecs and infrastructure to deliver 3D video. The second class of formats directly exploit the inter-view redundancy among multiview videos using inter-view prediction techniques, which is supported by the multiview video coding (MVC) extensions of the AVC video coding standard; similar extensions are also in the process of being specified in the context of the latest HEVC standard. Both of these formats include only the texture component of multiple views.

Another major category of 3D data formats are depth-based formats, which include depth information as part of the input data to facilitate the generation of intermediate views using depth image-based rendering (DIBR) techniques. A number of variants have been considered and evaluated, including the well-known 2D plus depth format. It has been found that an extended version of this format, referred to as multiview plus depth (MVD), which includes multiple view videos and their corresponding depth, provides better handling of disocclusions and is able to offer stereo compatibility. Efficient compression of the MVD format is now the subject of intense research; also standardization of the corresponding coding techniques is being evaluated and specified by the recently established Joint Collaborative Team on 3D Video Coding Extension Development (JCT-3V) of ITU-T WP3/16 and ISO/IEC JTC 1/SC 29/WG 11.

While the depth component undoubtedly increases the bitrate, it provides a 3D video coding system with useful geometric information of the scene. This geometric information can not only be used to generate intermediate view for display, but it can also be used to improve the prediction of the texture views using view synthesis prediction (VSP) techniques [1]. A conventional forward VSP normally incurs high complexity since it needs to warp an entire picture from one viewpoint to another before encoding the current picture; this process typically involves hole filling and blending steps. To reduce the non-negligible complexity at the decoder, backward VSP (BVSP) is considered as a possible solution, which obtains the block predictor only when BVSP is incurred. However,

BVSP requires the depth information of the current block prior to encoding the current block, which is a non-trivial problem under the typical assumption that texture is coded prior to depth. This problem is addresses with a novel method to derive the depth information from neighboring blocks with high accuracy. With the proposed approach, backward warping is enabled using depth information derived from either neighboring spatial (inter-view) compensated blocks or temporal compensated blocks. As a generalization, the proposed backward warping scheme is not only applied at the pixel level, but at the sub-block level as well.

The rest of the paper is organized as follows. Section 2 presents an overview of view synthesis prediction and its current status in the standard. In Section 3, a temporal and spatial backward view synthesis prediction (BVSP) is proposed under the framework of 3D-HEVC. Extensive experiments are conducted to verify the performance of the proposed BVSP scheme and the results will be discussed in Section 4. Finally the work is concluded in Section 5.

2. OVERVIEW OF VIEW SYNTHESIS PREDICTION

View synthesis prediction (VSP) is an effective approach to reduce the inter-view redundancy, utilizing the depth information to warp a reference view to the current view such that a predictor for the current view can be generated. When VSP was initially proposed to MVC [1], the depth information used for warping was estimated at the encoder and explicitly transmitted requiring additional rate. Of course, the depth could be estimated at the decoder, but this results in substantial and undesirable complexity increase at the receiver. In the context of MVD, depth information is already included as part of the input data, so it could be utilized directly for VSP.

2.1. Forward VSP

Traditionally, VSP is implemented in a forward warping manner, where the depth image from the reference view is used to warp the reference view to the current view. That is, the projection is conducted pixel-by-pixel from the reference picture to form a complete synthetic picture (stored in a reference picture buffer) before encoding or decoding the current picture. However, this process leads to significant complexity increase at the decoder since the warping is applied regardless of the usage of VSP for the current block. Moreover, such a projection is typically implemented with a loop over the pixel locations of a reference view. This may leave some pixels in the target view without any values, known as hole pixels because of disocclusions. Thus, a hole filling procedure is required before the synthetic picture is used as a reference, which requires the hole pixels to be identified and a new value assigned in a sequential order. Due to its irregularity, such a process is not friendly for parallel processing. Although forward VSP may be designed on a block basis [2], the warping of an oversized window in the reference view is required along with the hole filling process.

2.2. Backward VSP

To reduce the complexity of forward VSP, a block-based backward VSP (BVSP) is considered, where the depth information of the current block is inferred to determine the corresponding pixels in the reference picture. Such a projection is typically implemented with a loop over the pixel locations in the target view. In BVSP, all pixels will be assigned with a value during the projection process. Although some pixels may be incorrect if they are within a disocclusion area, our current design does not identify such pixels or correct their value since the synthesis results are only used as a predictor for compression and it is preferable to keep the complexity as low as possible.

Since texture is typically coded prior to depth, we propose a technique to estimate the depth of the current block using neighboring blocks from the reference view. The estimated depth block is then used for BVSP prediction. This process is described further in the following section.

3. PROPOSED BVSP SCHEME

In this section, a novel BVSP scheme is proposed by utilizing neighboring blocks to derive an estimated depth block, which is then used for BVSP in the context of 3D-HEVC.

3.1. Temporal and Spatial BVSP

Recall that the major obstacle of BVSP is the unavailability of the depth block corresponding to the current block when texture is coded prior to depth. However, with the available motion information of neighboring blocks, a depth block can be inferred by assuming the current block has the same motion vector as the neighboring block. In this way, the depth block to which the motion vector points to in the reference view can be used for backward warping in the current view.

Considering the compensation types of neighboring blocks, BVSP can be achieved using depth information derived from either neighboring spatial (inter-view) compensated blocks or temporal compensated blocks. Specifically, when a neighboring block is coded using an inter mode, there is an associated motion vector or disparity vector. Let us assume the motion vector (mv_x, mv_y) is identified from a neighboring block (Step 1 in Fig. 1), which points to a temporal reference picture at the time instance t, and the current block to be coded is at (x, y). It is assumed that both texture and depth image at time instance t have already been coded. Then we can locate a depth block at $(x+mv_x, y+mv_y)$ in the depth reference image (Step 2 in Fig. 1). The corresponding depth block is used as an estimate of the depth information for the current block. Backward warping is then performed

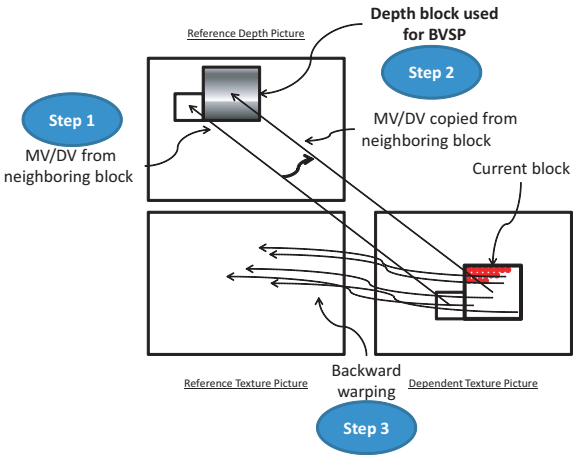

Fig. 1. Proposed BVSP

based on the estimated depth block (Step 3 in Fig. 1). When the motion vector points to a temporal reference picture, the BVSP is called as temporal BVSP herein. If the neighboring block uses a disparity vector instead of temporal motion vector, a depth block can be fetched from a interview depth reference image in a similar way, which could be then used for BVSP. As disparity vector points to a spatial (inter-view) reference picture, the BVSP is called as spatial BVSP herein. For anchor pictures, where no temporal prediction is used, only spatial BVSP can be applied to code a dependent view.

3.2. BVSP Mode Signaling

One way to signal the usage of the VSP mode is to have an adaptive flag at the block level, which is a good design in the context of an H.264/AVC based codec, where motion vector predictors (MVP) are derived by a median vector from the neighboring blocks. In HEVC, however, MVP derivation has been improved by introducing a merging candidate list, comprising of neighboring block motion vectors. An MVP index is then used to indicate the MVP to be used. To be inline with the HEVC design, it is proposed to include a new candidate index to signal the forward VSP mode in [3].

In this work, we reuse the indication method for the BVSP mode as that done in [3]. Once indicated, a synthetic reference block will be used instead of conventional prediction. As there are five spatial neighbors followed by temporal neighbors in HEVC, the new candidate is placed before the temporal neighbors and after the spatial neighbors in the list achieving the best RD performance.

The use of temporal and spatial BVSP may be indicated using high level syntax, e.g. at slice level. If only one of them is in use, a single merge candidate is inserted to the merging list. If both are in use, two new merge candidates need to be appended in the list.

3.3. VSP Warping and Motion Refinement

In the 3D-HEVC test model, there exists a tool to inherit neighboring blocks to derive a disparity vector predictor (DVP) and the derived disparity vector is then used to inherit motion prediction parameters from a reference picture. As the proposed BVSP mode can be regarded as an improved interview prediction mode, it is necessary to update the DVP derivation procedure by considering the newly introduced BVSP mode. In particular, if a neighboring block is coded in the BVSP mode, it is proposed to derive a DVP using a similar procedure as described in sub-section 3.1. A depth block is fetched as shown in Fig. 1 Step 2, and then the maximum depth from the depth block is converted to a disparity vector, and finally this derived disparity vector would be assumed as a refined disparity for the current block to do motion inheritance.

It should also be noted that the disparity vector derivation by referencing a depth block in depth reference picture could be further extended to other regular spatial or temporal neighboring blocks, as long as a depth block can be fetched from a depth image.

4. SIMULATIONS AND DISCUSSIONS

In this section, extensive experiments are conducted to verify the performance of the proposed BVSP scheme based on the lasted 3D-HEVC test model HTM5.1 [4]. Comparison is given from both the rate-distortion and in terms of computational complexity.

4.1. Simulation Conditions

The experiments are conducted following the common test conditions defined in [5], where there are three view points, each consisting of both texture and depth components. The coding condition is designed to characterize the applications which facilitates the intermediate view generation. In [5], the center view is coded as the base view and the other two views are coded as dependent views.

Test sequences from JCT-3V are classified into two categories. Three sequences are of XGA resolution (1024x768), and four others are of HD resolution (1920x1088). Among the seven test sequences, GT_Fly and Undo_Dancer are computer generated sequences using 3D models and their depths are ground truth. The depth maps for other sequences are estimated using stereo matching methods and contain some noise. For each sequence, four QP pairs, $(QP_{texture}, QP_{depth})$ are tested, including [(25, 34), (30, 39), (35, 42), (40, 45)]. In addition to coded video PSNR, synthesis PSNR is calculated between a synthetic picture using original texture and depth and another synthetic picture using the reconstructed texture and depth. Bjonteguard differences are calculated as an average coding gain between the proposal and the anchor.

	video 0	video 1	video 2	video PSNR / video bitrate	video PSNR / total bitrate	synth PSNR / total bitrate	enc time	dec time
Balloons	0.0%	-1.5%	-0.9%	-0.4%	-0.3%	-0.5%	101.7%	102.3%
Kendo	0.0%	-1.6%	-1.9%	-0.6%	-0.5%	-0.6%	103.3%	101.5%
Newspaper_CC	0.0%	-0.6%	-0.9%	-0.2%	-0.2%	-0.3%	103.7%	101.8%
GT_Fly	0.0%	-8.9%	-8.7%	-2.4%	-2.2%	-1.7%	101.2%	107.4%
Poznan_Hall2	0.0%	-0.4%	-2.8%	-0.6%	-0.5%	-0.6%	98.5%	102.6%
Poznan_Street	0.0%	-3.0%	-3.2%	-1.0%	-0.9%	-0.8%	102.0%	102.8%
Undo_Dancer	0.0%	-12.4%	-11.0%	-3.4%	-3.1%	-2.5%	98.5%	103.8%
1024x768	0.0%	-1.2%	-1.2%	-0.4%	-0.3%	-0.5%	102.9%	101.9%
1920x1088	0.0%	-6.2%	-6.4%	-1.8%	-1.7%	-1.4%	100.0%	104.1%
average	**0.0%**	**-4.1%**	**-4.2%**	**-1.2%**	**-1.1%**	**-1.0%**	**101.2%**	**103.2%**

Fig. 2. Simulation results on HTM5.1

Sequence	Pixel Usage
Balloons	0.82%
Kendo	1.53%
Newspapercc	0.41%
GhostTownFly	12.06%
PoznanHall2	1.34%
PoznanStreet	1.52%
UndoDancer	7.50%
Average	**3.60%**

Fig. 3. Pixel usage of BVSP

4.2. Simulation Results

In Fig. 2, spatial BVSP only is compared with HTM 5.1 anchor with a negative value indicating a bitrate saving. The average bitrate savings are summarized as follows: for video PSNR vs. video bitrate, bitrate saving is 1.2%; for coded video PSNR vs. total bitrate, the bitrate saving is 1.1%; and for synthesis PSNR vs. total bitrate, the bitrate saving is 1.0%. The decoding time is only increased by about 3%.

As part of our analysis, we found that the percentage of pixels using BVSP modes is about 3% on average shown in Fig. 3. In other words, the reported average gains could be achieved with relatively infrequent use of the BVSP modes. At the same time, it is noticed that the sequences with high quality depth would have more pixels coded using the proposed BVSP mode. For example, Undo_Dancer has about 7% pixels using BVSP mode and the gain over dependent view is above 10%. It is confirmed that the use of the BVSP modes is closely related with the gains for each sequence, and higher use/gains are achieved for sequences with better depth quality. From the above observations, we conclude that the proposed spatial BVSP mode work especially well for sequences with high quality depth.

Temporal BVSP is also tested on top of spatial BVSP. However, the additional gain is quite marginal under these test conditions. A major reason is that current test sequences contains less motion, and thus temporal BVSP mode is less competitive with traditional temporal prediction modes. Temporal BVSP needs to be further evaluated on sequences with high motion.

4.3. Complexity Discussions

With the proposed BVSP scheme, the main impact on a codec design is related to the motion compensation (MC) module and memory bandwidth.

In the conventional MC module, a single vector is used to address a predictor block in the reference image. With the proposed BVSP method, each pixel may have a different vector requiring individual addressing to pick up a predictor sample. This would increase the time for addressing and data fetching. In order to mitigate such problems, sub-PU level (e.g. 2x2, 4x4) BVSP are also evaluated, where each sub-PU block uses the same vector to get the predictors. It is found that the bitrate saving only drops about 0.1% compared to pixel-wise BVSP while the addressing time is reduced to a quarter if 2x2 sub-PU BVSP is used. With BVSP at 4x4 sub-PU level, the rate savings reduces by 0.2% compared to pixel-wise BVSP and the addressing time is reduced to 1/16th. Alternatively, an over-sized predictor block defined by a minimum and maximum disparity of a BVSP block may be fetched in one cycle. Though the fetched data size is increased a bit, it could be done within one fetching cycle using SIMD.

A second impact is the bandwidth required to fetch a depth block from the depth reference picture buffer. Based on the analysis in subsection 4.2, the coding benefits is well correlated to the increased bandwidth, so there is a trade-off between implementation cost and expected performance improvements.

5. CONCLUSIONS

In this paper, a novel temporal and spatial backward view synthesis prediction method using motion and disparity from neighboring blocks is proposed. In particular, we presented the integration of the BVSP mode into 3D-HEVC, and note that this method has been recently adopted to the test model for standardization. Additionally, we describe the use of depth information not only for VSP, but for motion refinement as well. Extensive simulations demonstrate that the spatial BVSP mode can provide a notable gain with minimal complexity and impact on codec design, while the temporal BVSP needs to be further evaluated with more varied contents to verify its effectiveness.

6. REFERENCES

[1] E. Martinian, A. Behrens, J. Xin, and A. Vetro, "View synthesis for multiview video compression," in *Picture Coding Symposium*, Apr. 2006, vol. 37, pp. 38–39.

[2] Y. Zhang, Y. Zhao, and L. Yu, "3D-CE1.h: Forward warping block-based view synthesis prediction," in *Joint Collaborative Team on 3D Video Coding Extension Development of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, JCT3V-C0087*, Geneva, Swiss, Jan., 2013.

[3] F. Zou, D. Tian, and A. Vetro, "View synthesis prediction using skip and merge candidates for HEVC-based 3D video coding,"

in *Circuits and Systems (ISCAS), 2013 IEEE International Symposium on*, May 2013.

[4] G. Tech, K. Wegner, Y. Chen, and S. Yea, "3D-HEVC test model 2," in *Joint Collaborative Team on 3D Video Coding Extension Development of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, JCT3V-B1005*, Shanghai, China, Oct.,2012.

[5] D. Rusanovskyy, K. Mueller, and A. Vetro, "Common test conditions of 3DV core experiments," in *Joint Collaborative Team on 3D Video Coding Extension Development of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, JCT3V-B1100*, Shanghai, China, Oct.,2012.