EFFICIENT EDGE-PRESERVING INTERPOLATION AND IN-LOOP FILTERS FOR DEPTH MAP COMPRESSION

Viet-Anh Nguyen and Dongbo Min

Advanced Digital Sciences Center, Singapore

Minh N. Do

University of Illinois at Urbana-Champaign

ABSTRACT

Due to abrupt signal changes on object boundaries, a depth video compressed by conventional video coding standards often introduces serious coding artifacts over the boundaries, which severely affect the quality of a synthesized view. In this paper, we propose an edge-preserving depth interpolation filter based on weighted mode filtering to provide more accurate fractional-pixel samples in the motion-compensated interpolation for an effective inter-coding of the depth video. In addition, an efficient post-processing method is also proposed to further suppress the coding artifacts on the depth video and utilized as an in-loop filter. Experimental results show the proposed methods can significantly improve the synthesized view quality in terms of both objective and subjective measures.

Index Terms— Weighted mode filtering (WMF), depth coding, 3D video, motion-compensated interpolation

1. INTRODUCTION

Multiview video representations have attracted much attention recently in new applications such as 3-dimensional TV (3DTV) or freeview point TV (FTV). While 3DTV aims to provide users with 3D depth perception, FTV can give users the freedom of selecting a viewpoint according to a user preference [1]. For successful development of 3D video systems, many technical issues should be resolved from capturing and analyzing the multiview images, compressing and transmitting the data, and rendering multiple images on various 3D displays.

Essentially, the performances of 3DTV and FTV are often governed with an increase number of multiple views, which results in a huge amount of data. For efficient storage and transmission, a popular multiview video plus depth data format has been proposed to realize such 3D video systems [2]. Such a format requires color videos transmitted together with associated depth maps for a limited number of views, which can be used to synthesize arbitrary virtual views at the receiver. While a number of methods have been proposed to efficiently compress multiview video, depth video coding has not been studied extensively.

In general, depth video coding aims to reduce a bit rate as much as possible while ensuring the quality of the synthesized view, not the depth map itself. In addition, depth video is typically characterized by homogeneous regions partitioned by sharp edges, which should be preserved in order to provide a high-quality synthesized view. Thus, the straightforward compression of depth video using the existing video coding standards such as H.264/AVC may cause serious coding artifacts along the depth discontinuities, which ultimately affect the synthesized view quality.

To alleviate the above problem, many techniques have been proposed to compress the depth video by taking into account its unique characteristics [3, 4, 5]. These techniques often focus on an efficient representation of depth edge regions to preserve the object boundaries in the reconstructed depth video. In addition, post-processing techniques can also be utilized to suppress the coding artifacts along object boundaries to obtain a better quality of the synthesized view [6, 7, 8].

In this paper, we propose novel coding tools for efficient depth compression on the framework of a conventional video coding standard. In particular, we utilize a weighted mode filtering (WMF) proposed in our previous work [9] to design an edge-preserving depth interpolation filter that provides more accurate fractional-pixel samples in order to achieve better sub-pixel motion-compensated prediction. To our best knowledge, no similar work has been reported in the literature to address the problem of inaccurate sub-pixel motion-compensated interpolation for depth compression. Furthermore, an efficient post-processing method is also utilized as an in-loop filter to further suppress the coding artifacts.

The remainder of the paper is organized as follows. Section 2 briefly provides the overview of weighted mode filtering. Section 3 presents the proposed edge-preserving interpolation and in-loop filters. Experimental results are shown in Section 4. In Section 5, we conclude the paper by summarizing the main contributions.

2. WEIGHTED MODE FILTERING

Weighted mode filtering was introduced in [9] to enhance the depth map acquired from depth sensors. In this paper, we utilize such a filter in an effective manner in the context of depth compression. For completeness, we provide here a brief redefinition of the weighted mode filtering based on the localized histogram concept.

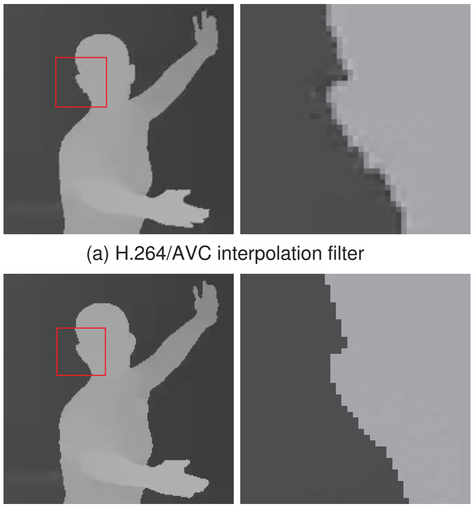
A localized histogram $H(p,d)$ for a reference pixel p and d^{th} bin is computed using a set of its neighboring pixels inside a window, which was introduced by Weijer *et al.* [10]. Specifically, given a discrete function of depth signal $D(p)$ whose value ranges 0 to $L-1$, the localized histogram $H(p,d)$ is defined at the pixel p and d^{th} bin ($d \in [0, L-1]$). The localized histogram means that each bin has a likelihood value which represents an occurrence of neighboring pixels q inside rectangular (or any shape) regions. The likelihood value is measured by adaptively counting a weighting value computed with a kernel function $w(p,q)$ as

$$H(p,d) = \sum_{q \in N(p)} w(p,q) G_r(d - D(q)) \,, \qquad (1)$$

where $w(p,q)$ is a non-negative function which defines the correlation between the pixels p and q. $N(p)$ is the set of neighboring pixels in a window centered at p. A spreading function G_r models

This study is supported by the research grant for the Human Sixth Sense Programme at the Advanced Digital Sciences Center from Singapore's Agency for Science, Technology and Research (A*STAR).

978-1-4673-2533-2/12/$26.00 ©2012 IEEE



(a) H.264/AVC interpolation filter

(b) Proposed interpolation filter

Fig. 1. Samples at quarter-pixel positions obtained by the H.264/AVC 6-tap and bilinear interpolation filters and the proposed interpolation filter.

errors that may exist on the input data $D(p)$. The weighting function represents the influence of the neighboring pixels on the localized histogram. In essence, the neighboring pixel which exhibits a stronger correlation with the reference pixel p has a larger weighting value $w(p,q)$. The final filtered solution $\hat{D}(p)$ is calculated by

$$\hat{D}(p) = \arg\max_{d} H(p,d). \tag{2}$$

We call this histogram-based approach a weighted mode filtering, in which the final solution is obtained by seeking the highest mode of the weighted distribution $H(p,d)$.

3. PROPOSED ALGORITHMS

3.1. Edge-preserving interpolation filter

In the existing video coding standard such as H.264/AVC, one-dimensional 6-tap interpolation and bilinear interpolation filers are used to obtain the samples at fractional-pixel positions for sub-pixel motion-compensated prediction. Such interpolation filters are designed for the color video, which are not suitable for the depth map containing sharp edges. They not only produce the blurry edges, but also spread the coding artifacts presented in the reconstructed samples at full-pixel positions. Fig. 1(a) illustrates such quarter-pixel samples obtained by the H.264/AVC interpolation filters, in which the region highlighted in red is magnified for better visualization. Due to these blurry edges, the sub-pixel motion-compensated prediction is not optimally matched with the block in the current frame containing sharp edges. This will result in undesirable high residues along abrupt sharp edges. As a result, not only more bits are required to code the residues, but also serious ringing artifacts will be introduced in the compressed depth map due to the quantization of the high-frequency information and severely affect the synthesized view quality.

Inspired by the above observation, we propose in this paper an efficient interpolation filter to obtain more accurate depth samples at fractional-pixel positions in order to achieve better sub-pixel motion-compensated prediction. In particular, we aim to obtain these fractional-pixel samples while preserving the edge information by utilizing the weighted mode filtering. In general, the compressed depth map is often transmitted together with the associated color video in order to synthesize the virtual view at the receiver side. As two correlated depth pixels along the depth boundaries usually exhibit a strong photometric similarity in the corresponding video pixels, we utilize the color video pixels $I(p)$ as the guided information for the depth interpolation. Specifically, the weighting value $w(p,q)$ in Eq. (1) is represented by the range filter of the color video G_I and the localized histogram can be written as

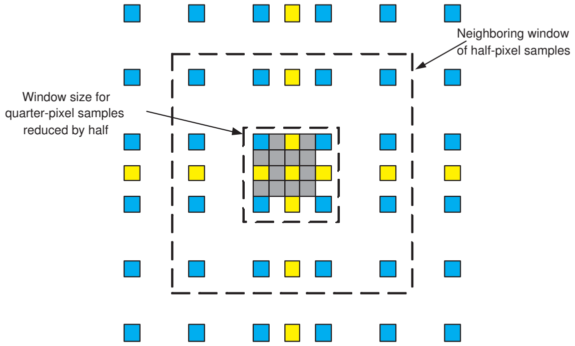

Neighboring window of half-pixel samples
Window size for quarter-pixel samples reduced by half

Fig. 2. Filtering for fraction-pixel accurate motion compensation. Example sparse samples at the full-pixel grid, half-pixel grid, and quarter-pixel grid are highlighted in cyan, yellow, and gray colors respectively.

$$H(p,d) = \sum_{q \in N(p)} G_I(I(p) - I(q)) G_r(d - D(q)) \,. \tag{3}$$

where both G_I and G_r are chosen as Gaussian filters. Using the Gaussian filter G_r to model depth errors will diminish faster the influence of the strong neighboring outliers by down-weighting, while still incorporating the influence of the neighboring inliers.

Here, we propose to obtain the half-pixel and quarter-pixel samples by applying the weighted mode filtering at the fractional-pixel grids using Eqs. (2) and (3). Initially, the samples at the half-pixel grid are obtained. In this step, we only consider the neighboring pixels $q \in N(p)$ at full-pixel positions (e.g., samples in cyan as shown in Fig. 2) to compute the localized histogram. The half-pixel samples are then obtained by using Eq. (2). In the second step, the samples at the quarter-pixel grid are then obtained by considering the neighboring pixels at both full-pixel and half-pixel positions (e.g., cyan and yellow samples in Fig. 2) for the computation of $H(p,d)$. To obtain the guided color information at half-pixel positions, the simple bilinear interpolation can be utilized. Note that the size of the neighboring window $N(p)$ is reduced by half in the second step so that the same number of neighboring pixels is considered in each step.

Fig. 1(b) shows the quarter-pixel samples obtained by the proposed interpolation filter. As can be seen from the figure, by using the mode operation on the localized histogram, the proposed interpolation filter can reduce an unnecessary blur along the object boundaries. In addition, utilizing the structural similarity between the color and depth videos through the guided color information not only reduces the coding artifacts, which may be spread from the compressed full-pixel samples, but also attains more accurate depth edge information by aligning with the color video.

3.2. In-loop filter

Note that the proposed interpolation filter can provide a better motion-compensated interpolation for the inter-coded region. Serious coding artifacts still exist along sharp edges in the intra-coded region. Moreover, since the existing in-loop filter (e.g., H.264/AVC

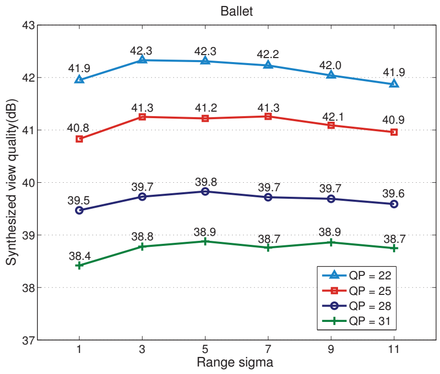


Fig. 3. PSNR (dB) results of the synthesized view obtained by encoding the depth video at different QPs using the proposed in-loop filter with different values of range sigma σ_r.

deblocking filter) is mainly designed to reduce the blocking artifacts for color videos, it is inefficient in removing ringing artifacts in the compressed depth video.

To further improve the depth coding performance, we propose to employ the weighted mode filtering on the reconstructed depth map to suppress the remaining coding artifacts. Similar to Section 3.1, the color information is used to guide the depth denoising process as described in Eq. (3). The filtering process only considers the full-pixel samples and is utilized as an in-loop filter to replace the conventional deblocking filter. As the spreading function G_r is used to model the errors on the input data, the Gaussian parameter σ_r should be determined by the amount of noise in the depth data. To select an optimal value of σ_r, we compressed the depth video at different QPs using the proposed in-loop filter in the encoder with different values of σ_r. An objective performance is measured indirectly by analyzing the quality of the synthesized view (refer to Section 4 for the simulation setup details). Fig. 3 shows the peak-signal-to-noise ratio (PSNR) of the synthesized view obtained using different values of σ_r. The results show that with different amounts of noise introduced by the quantization artifact, setting σ_r to 3 generally provides the best synthesized view quality while maintaining low computational complexity.

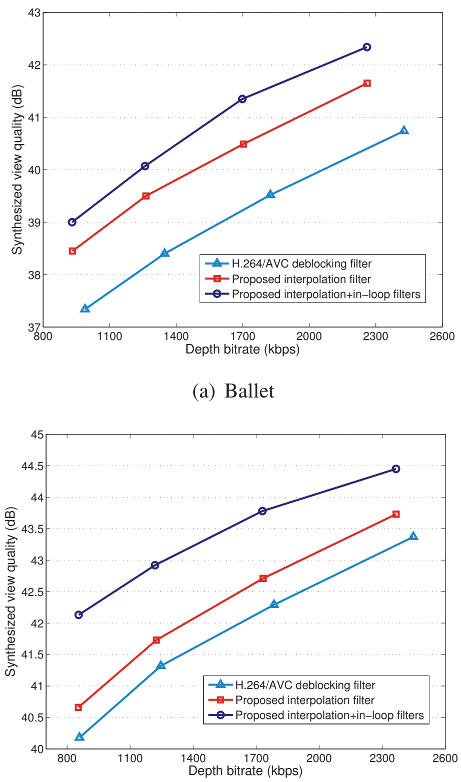
It should be noted that both color and depth information is used to synthesize the virtual view. Thus, using the color information to guide the depth denoising process intuitively can achieve a better synthesized view quality as shown later in the experimental results.

4. EXPERIMENTAL RESULTS

We have conducted a series of experiments to evaluate the performance of the proposed depth compression techniques. We have tested with the Breakdancers and Ballet test sequences with resolutions of 1024×768, of which both the color video and depth map are provided from Microsoft Research [11].

The experiments were conducted by using the H.264/AVC Joint Model Reference Software JM17.2 to encode the depth map of each view independently. For each test sequence, we encoded two (left and right) views for both color and depth videos using various QPs.

To measure the performance of the proposed methods, we analyzed the quality of the color information for the synthesized intermediate view. Among 8 views, view 3 and view 5 were selected as reference views and a virtual view 4 was generated using the View Synthesis Reference Software (VSRS) 3.0 provided by MPEG [12]. For an objective comparison, the PSNR of each virtual view generated using compressed depth maps was computed with respect to that generated using the original depth map. Rate distortion (RD) curves were obtained by the total bit rate required to encode the depth maps of both reference views and the PSNR of the synthesized view.



(a) Ballet

(b) Breakdancers

Fig. 4. RD curves obtained by encoding the depth maps using the different methods.

To evaluate the performance of the proposed interpolation filter, we compared with the existing interpolation filters in H.264/AVC. For fair comparison, the H.264/AVC deblocking filter was used in both methods. Fig. 4 shows the RD curves obtained by these methods. Not surprisingly, the proposed interpolation filter achieved a better synthesized view quality compared with the existing filters, since an improved motion-compensated interpolation could be obtained using better fractional-pixel samples with sharp and precise edge information. Specifically, we achieved about 1.2-dB and 0.5-dB gain in PSNR of the synthesized view quality in terms of average Bjontegaard metric [13] for the Ballet and Breakdancers sequences, respectively. As discussed in Section 3.1, more accurate motion-compensated prediction reduced the number of bits required to code the residues while lessening the coding artifacts. However, it is observed that due to fast motion and temporal inconsistency, many blocks in the Breakdancers sequence were intra coded and not benefited from the proposed interpolation filter; leading to a slightly lower PSNR gain compared with the Ballet sequence. For visualization, Figs. 5 and 6 show the reconstructed depth maps and the corresponding synthesized views obtained by different methods. It was evident that using the proposed interpolation filter improved the quality of the inter-coded regions in the reconstructed depth map with reduced ringing artifacts compared with that of H.264/AVC (see the head region in Fig. 5(a)-(b)). Meanwhile, serious coding

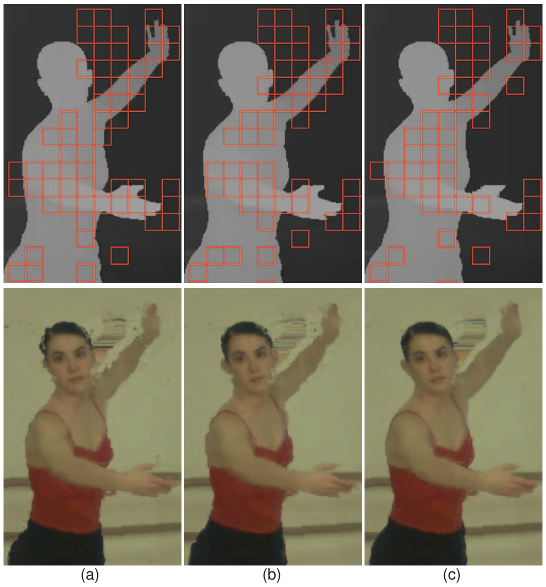


Fig. 5. Sample frames of the reconstructed depth maps and rendered views for the Ballet sequence obtained by different encoding schemes: (a) H.264/AVC with deblocking filter, (b) Proposed interpolation filter with the deblocking filter, (c) Proposed interpolation and in-loop filters. The red color blocks indicates the intra-coded regions in the depth map.

artifacts remained in the intra-coded regions highlighted in red color and severely affected the rendering view quality.

As the in-loop filter is proposed to further suppress the remaining coding artifacts, we evaluated the performance by encoding the depth videos using both the proposed interpolation and in-loop filters. The RD results are also shown in Fig. 4 and the sample frames are provided in Figs. 5 and 6. The results show that by using the proposed in-loop filter, the synthesized view quality was further improved by about 0.67-dB and 1.21-dB Bjontegaard gains in PSNR for the Ballet and Breakdancers sequences, respectively, compared with that obtained by using the proposed interpolation filter with the conventional deblocking filter. This is because unlike the deblocking filter, the proposed in-loop filter efficiently suppressed the coding artifacts remained in the reconstructed depth map, especially in the intra-coded regions with sharp edges (see the reconstructed depth map in Fig. 5(c) and Fig. 6(c)). The proposed method not only resulted in sharp edges with reduced coding artifacts, but also preserved the object structures by utilizing the structural similarity between the color and depth videos. As a result, the synthesized view quality was not only improved in terms of PSNR, but also subjectively better, especially around the object boundaries.

5. CONCLUSION

We have presented in this paper novel coding tools for efficient depth compression using the existing video coding standard. Specifically, we have proposed an edge-preserving depth interpolation filter using the weighted mode filtering to obtain more accurate fractional-pixel samples in the motion-compensated interpolation. In addition, an efficient denoising in-loop filter has also been proposed and replaced the conventional deblocking filter to further suppress the coding artifacts. Experimental results have shown that our proposed filter not only provided a significant PSNR gain over the synthesized view quality, but also resulted in a better subjective quality compared with the conventional H.264/AVC.

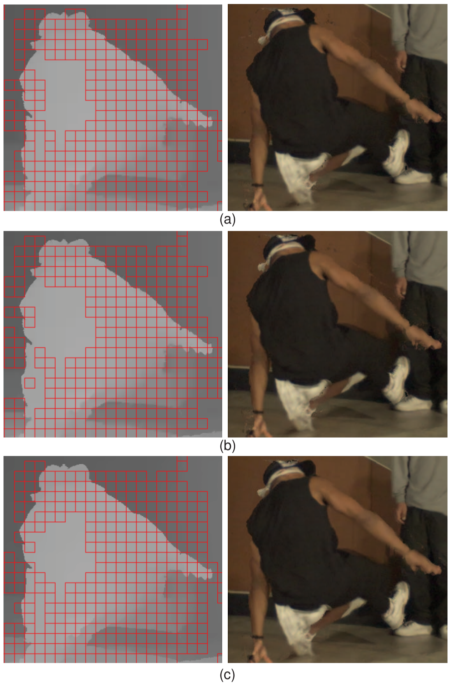


Fig. 6. Sample frames of the reconstructed depth maps and rendered views for the Breakdancers sequence obtained by different encoding schemes: (a) H.264/AVC with deblocking filter, (b) Proposed interpolation filter with the deblocking filter, (c) Proposed interpolation and in-loop filters. The red color blocks indicates the intra-coded regions in the depth map

6. REFERENCES

[1] D. Min, D. Kim, S. Yun, and K. Sohn, "2D/3D freeview video generation for 3DTV system," *Signal Processing: Image Communication*, vol. 24, no. 1-2, pp. 31-48, 2009

[2] P. Merkle, A. Smolic, K. Muller, and T. Wiegand, "Multi-view video plus depth representation and coding," in *Proc. IEEE ICIP*, 2007.

[3] Y. Morvan, P. With and D. Farin, "Platelet-based coding of depth maps for the transmission of multiview images," in *Proc. of SPIE, Stereoscopic Displays and Applications*, vol. 6055, pp. 93-100, 2006.

[4] G. Shen, W.-S. Kim, A. Ortega, J. Lee, and H. Wey, "Edge-aware Intra Prediction for Depth-map Coding," in *Proc. IEEE ICIP*, 2010.

[5] W.-S. Kim, A. Ortega, P. Lai, D. Tian, and C. Gomila, "Depth map distortion analysis for view rendering and depth coding," in *Proc. IEEE ICIP*, 2009.

[6] P. Lai, A. Ortega, C. C. Dorea, P. Yin, and C. Gomila, "Improving View Rendering Quality and Coding Efficiency by Suppressing Compression Artifacts in Depth-Image Coding," In *Proc. SPIE VCIP*, 2009.

[7] K.-J. Oh, A. Vetro, and Y.-S. Ho, "Depth Coding Using a Boundary Reconstruction Filter for 3-D Video Systems," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 21, no. 3, pp. 350-359, 2011.

[8] S. Liu, P. Lai, D. Tian, and C. W. Chen, "New Depth Coding Techniques With Utilization of Corresponding Video," *IEEE Trans. on Broadcasting*, vol. 57, no. 2, pp. 551-561, 2011.

[9] D. Min, J. Lu, and M. N. Do, "Depth Video Enhancement Based on Weighted Mode Filtering," *IEEE Trans. on Image Processing*. (to appear)

[10] J. Weijer and R. Boomgaard, "Local Mode Filtering," in *IEEE Proc. Computer Vision and Pattern Recognition*, pp. 428-433, 2001.

[11] *MSR 3-D Video Sequences* [Online]. Available: http://www.research.microsoft.com/vision/ImageBasedRealitites/3DVideoDownload.

[12] M. Tanimoto, T.Fujii, and K. Suzuki, "View synthesis algorithm in view synthesis reference software 3.0 (VSRS3.0)," Tech. Rep. Document M16090, ISO/IEC JTC1/SC29/WG11, Feb. 2009.

[13] "An excel add-in for computing Bjontegaard metric and its evolution," document VCEG-AE07, ITU-T SG16 Q.6, Jan. 2007.