CONTENT ADAPTIVE IN-LOOP DEPTH MAP FILTER FOR HEVC BASED 3DV CODING

Jianqiang He[1], Siwei Ma[1], Nan Zhang[2] and Wen Gao[1]

[1]Institute of Digital Media, Peking University, Beijing, 100871, China
[2]School of Biomedical Engineering, Capital Medical University, Beijing, China
{jqhe, swma, wgao}@pku.edu.cn, zhangnan@ccmu.edu.cn

ABSTRACT

In this paper, a content adaptive in-loop depth map filter is proposed for HEVC based 3DV video coding to improve the quality of the synthesized views. The proposed depth map filtering scheme is block based by reusing the TU (transform unit) split structure in HEVC. Firstly, TU are classified into 3 categories: flat, directional, and textureless, according to the characteristics of the depth map. Then four kinds of filters are designed by considering the characteristics of the current TU and neighbor TUs jointly. The proposed scheme is incorporated into HTM3.0- the reference software of HEVC based 3D video coding. The experimental results show both subjective and objective quality improvement. The average BD-rate reduction is 0.65% for synthesis views in HD test sequences, and the maximum BD-rate reduction is up to 1.6%.

Index Terms**—** HEVC-3DV, Depth map, In loop filter

1. INTRODUCTION

With the development of 3D display technologies such as FTV, 3DV has gained much more attentions. MPEG started the 3DV activity to establish an efficient coding standard for 3D video coding in 2011, in which "video+depth" scheme is adopted for multi-view generation by view synthesis [1]. Depth map is used to synthesize the intermediate view video while not be visible for viewer. Using of depth map has two advantages: first, it can significantly reduce the transmitted view video data; second, the receiver side can synthesize arbitrary viewpoint video with the aid of depth. So providing a high-quality depth map is important to 3D perception quality.

In the current block based prediction/transform hybrid coding framework, video coding actually uses a greedy strategy to decide the mode of each block individually. However the optimal performance of the each block cannot guarantee the optimal coding performance of the whole video sequence. In H. 264/AVC and the state-of-the-art HEVC, deblocking filter is designed to suppress blocking artifacts and enhance both the subjective and objective quality. It is used at block boundary in H.264 [2] or CU/PU/TU boundary in HEVC [3]. However, depth maps are mainly characterized by sharp edges and large areas of nearly constant or slowly varying sample values, and the real depth object edge often do not coincide with the block boundary in depth map. So the current deblock filter is not suitable for depth map anymore. Firstly, deblocking filter is always performed along the direction perpendicular to the block boundary, without considering the real edge of the depth image inside the block. Secondly, the same filter orientation and thresholds are applied to all images without considering the local characteristics of the depth image. Some depth map filtering methods, such as trilateral loop filtering [4], adaptive bilateral filter [5] and region-based anisotropic filter [6] were proposed to keep the sharp edge and supress the noise in bilateral side of the edge. But those methods are mainly based on frame level, and didn't consider well either the characteristics of artifacts in TU brought by quantization or the correlation of distortion between neighbour TUs.

In this paper, we focus on improving the synthesized views quality by introducing a new depth map filter. A content adaptive in-loop depth map filter is proposed for HEVC based 3DV video coding to preserve depth map details and improve the subjective and objective quality of the synthesized views. We reuse the TU split structure in HEVC, and classify TU into 3 categories according to the characteristics of the depth map, then decide both boundary strength (BS) and filter type adaptively by considering well of the correlation between neighbor TUs and the current TU.

The rest of this paper is organized as follows: In Section 2, HEVC-3DV and depth coding are reviewed briefly. In Section 3, a content adaptive in-loop filter for depth map is presented. Section 4 describes the experiments and results analysis. Section 5 draws the conclusion.

2. DEPTH CODING IN HEVC-3DV

In the state-of-the-art 3DV codec HEVC-3DV [1], 3D video is represented using the Multiview Video plus Depth (MVD) format. Only 3 views (like V_0 V_1 V_2) as well as associated depth maps are coded, and additional intermediate views (usually 6 synthesized views $V_{0.25}$ $V_{0.5}$ $V_{0.75}$ $V_{1.25}$ $V_{1.5}$ $V_{1.75}$) suitable for auto-stereoscopic display can be synthesized using depth-image-based rendering (DIBR) techniques after decoding the video and depth data.

978-1-4799-0356-6/13/$31.00 ©2013 IEEE

For the coding of depth maps, basically the same as texture video, however, some tools like in-loop filter are generally disabled and some tools are modified such as the mode decision metrics of depth block is changed from RDO to VSO (View Synthesis Optimization). In 3DV coding, the lossy coding of depth data causes distortions in the synthesized intermediate views. The efficiency of depth coding is improved by considering this property. Unlike AVC-3DV [7], in HEVC-3DV, the synthesized view distortion change (SVDC) is used as distortion measure for the mode decision process for depth maps, and using synthesized views derived from the original texture and depth as the anchor.

The adaption of VSO improves the RD performance of the synthesised view dramatically. VSO strategy selects the mode which gets the optimal RD performance of each individual block in synthesised view corresponding to the current depth block. But it is local optimal, couldn't guarantee global optimal in all the synthesised video sequences. And in HEVC-3DV, the QP of depth map is higher than the corresponding texture view by 5 to 9, which cause great degradation to the depth quality especially in I frames. This degradation error will diffuse to the following coding blocks in current frame or blocks in other frames which refer to this degradation depth block. That would bring error in synthesized frames and degrade the 3D perception quality. So it needs in-loop filtering for depth map in accordance with the depth map characteristics, to improve the subjective and objective quality of synthesized views.

3. CONTENT ADAPTIVE IN-LOOP FILTER FOR DEPTH MAP

In HEVC-3DV, TU is the quantization block unit. And the distortion of a block mainly comes from the quantization of the transformed residual. When the QP is high, high frequency components in a TU will be discarded while the DC component changed litter. Actually, the DC component of a TU is mainly preserved in prediction unit. This will help design a new depth in loop filter.

According to the above analysis, we establish the following hypothesis: 1) in most case, the content of TU is flat or including only one edge in TU of I frames in depth map; 2) the DC component of a TU in Intra mode is changed litter between the original and the reconstruction content.

In this section, we present a new depth in loop filter, which classifies TU into 3 categories and decides both BS and filter type adaptively by considering well of the correlation between neighbor TUs and the current TU.

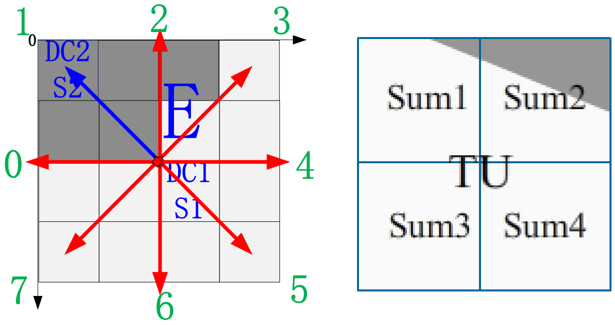
3.1. TU classification

In HEVC-3DV, TU is a small granularity (size from 4x4 to 32x32) and simple construction block unit in I frames of depth map, so we can classify TUs into 3 categories by the characteristics of TU (denotes as TU_Dir): “flat”, “directional” and “textureless”.

Firstly, after binarization processing, TU is divided into two (or one) parts, the area of the larger part denotes as S1, and the average depth grey value of S1 denotes as DC1; while parameter of the other part are S2 and DC2, where $S1 \geqslant S2$. If the TU is flat, then S2=0. So we can describe one TU with a parameter set {TU_Dir, DC1, S1, DC2, S2}, as showed in fig 1.(a). Then the average distortion of each pixel introduced by binarization processing is calculated as below.

$$\overline{Dist\_B} = \frac{1}{S1+S2}\left(\sum_{i=0}^{S1-1}|P(i)-DC1| + \sum_{j=0}^{S2-1}|P(j)-DC2|\right) \qquad (1)$$

Where $P(i)$ and $P(j)$ are the pixel value in $S1$ and $S2$ respectively. If $\overline{Dist\_B} > Thres\_B$ (set as 5), means the TU doesn't satisfy the hypothesis, and defines TU_Dir as “textureless”. If $S2 = 0$, defines TU_Dir as “flat”. Otherwise, TU is “directional” (TU_Dir from: 0, 1, 2…7).



(a) (b)

Fig 1. (a)TU directions (0,1...7). (b)Divide TU into four equal parts and accumulate pixels of each part.

There are many methods to obtain the direction of an area, in [8] adapting orientation energy edge detection, in [9] using the intra mode to indicate the direction. But these methods are complex and mean to get the direction of the block boundary not the object edge in depth block; there are not suitable for depth map. As TU is a small and simple block, we design a simple algorithm to get a rough direction.

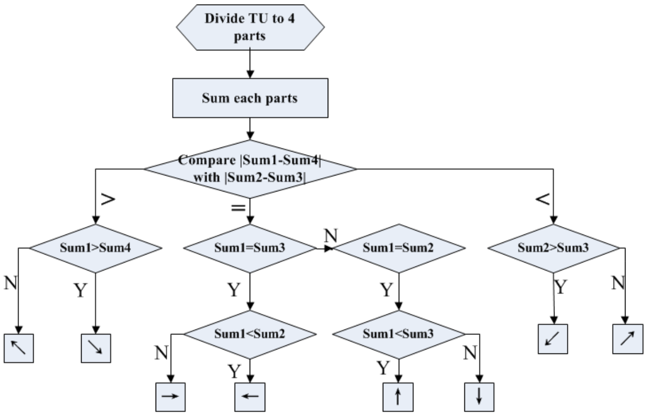
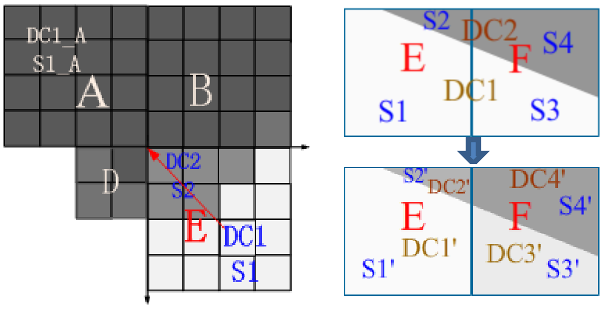


Fig 2. Flow diagram of get TU_Dir (as the arrows)

First, TU is divided into four equal parts, and pixels of each part is accumulated, denote as Sum1, Sum2, Sum3, and Sum4, as shown in fig 1.(b). Then for the case $\mathrm{DC1} > DC2$, we get the TU_Dir as in the flow diagram in Fig.2. For the case $\mathrm{DC1} < \mathrm{DC2}$, the process is similar, just reverse the direction. However to make the direction decision more robust, we adjust the judgment condition by introducing Tolerate_err , and replace judgment condition $|\mathrm{Sum1} + \mathrm{Sum2}| = |\mathrm{Sum3} + \mathrm{Sum4}|$ with $|(\mathrm{Sum1} + \mathrm{Sum2}) - (\mathrm{Sum3} + \mathrm{Sum4})| \le$ Tolerate_err(where Tolerate_err =128xTU_width).

3.2. Deciding filter type

After getting the parameter set {TU_Dir, DC1, S1, DC2, S2} of each TU, in loop filter decision is made by considering the relationship between the adjacent TUs and the current TU.



(a) (b)

Fig 3. (a) TU_Dir point to flat TU. (b) Edge cross adjacent TUs and after transform-quantization, changes the DC1 DC2 value of each part.

Borrowing from H.264 and HEVC, we introduce BS to indicate filter types. In former codecs, each block boundary has a BS; while in proposed depth map filter, each TU has only one BS. Denotes TU_Dir_N as the TU_Dir of block N, the rest may be deduced by analogy. The algorithm description of BS decision is given below.

```
1.  If TU_Dir is textureless:
        BS=0.
2.  else if TU_Dir is flat, and there are two adjacent flat TUs around(such as TU A and TU B), and |DC1_A – DC1_E| <4:
        BS=4.
3.  else (TU_Dir is one of eight directions, like fig 1.(a))
    3.1  If the TU pointed by the TU_Dir, like A in fig 3.(a), is flat and larger, and |DC1_A – DC2_E| ≤8:
             BS=3.
    3.2  else if as shown in fig 3.(b), if TU_Dir of the current TU E is equal to the TU_Dir of adjacent TU F; and parameters DC1, DC2 in two TUs satisfied a certain conditions:
             BS=2.
    3.3  else if the opposite side of the TU_Dir is a flat TU such as TU O, and TU O is much larger than the current TU E, in addition |DC1_O – DC1_E| <5:
             BS=1.
4.  Otherwise: BS=0.
```

3.3. Depth TU in loop filter

After getting the BS of each TU, the TU will be filtered based on BS value.

For BS=4, means the current TU E is flat, and there is a large flat area around TU E, and the depth value is close between two areas. Then the current TU will be filtered as:

$$DC1\_E = \frac{DC1\_A \times S1\_A + DC1\_B \times S1\_B + DC1\_E \times S1\_E}{S1\_A + S1\_B + S1\_E} \quad (2)$$

For BS=3, means the current TU pointing to a large flat area, and the depth value is close, as shown in fig 3.(a).

$$DC2\_E = Clip3(DC2\_E - 8, DC2\_E + 8, \frac{DC1\_A \times S1\_A + DC2\_E \times S2\_E}{S1\_A + S2\_E}) \quad (3)$$

For BS=2, according to the hypothesis after the quantization, the DC component of an intra-block is nearly unchanged, as shown in fig 3.(b). s.t:

$$DC1 \times S1 + DC2 \times S2 \approx DC1' \times S1' + DC2' \times S2' \quad (4)$$

$$DC1 \times S3 + DC2 \times S4 \approx DC3' \times S3' + DC4' \times S4' \quad (5)$$

Where $S1 \approx S1', S2 \approx S2', S3 \approx S3', S4 \approx S4'$

Approximate solution of DC1 and DC2 can be solved out from (4) (5). Denotes $S\_cross = S2 \times S3 - S1 \times S4$

If S_cross =0:

$$DC1 = \frac{DC1' \times S1 + DC3' \times S3}{S1 + S3} \quad (6)$$

$$DC2 = \frac{DC2' \times S2 + DC4' \times S4}{S2 + S4} \quad (7)$$

Otherwise:

$$DC1 = \frac{S2 \times S4 \times (DC2' - DC4') + S1 \times S4 \times DC1' - S2 \times S3 \times DC3'}{S\_cross} \quad (8)$$

$$DC2 = -\frac{S1 \times S3 \times (DC1' - DC3') + S2 \times S3 \times DC4' - S1 \times S4 \times DC2'}{S\_cross} \quad (9)$$

Then, DC1 and DC2 will be clipped.

$$DC1 = Clip3(DC1 - 5, DC1 + 5, DC1) \quad (10)$$

$$DC2 = Clip3(DC2 - 8, DC2 + 8, DC2) \quad (11)$$

For BS=1, is similar to the case BS=3, except TU A is at the other side (opposite of current TU direction).

Finally, the pixels in S1 and S2 will be replaced with the modified DC1 and DC2 respectively. For any other cases, the depth TU block will not be filtered, and the filter is only carried out in I frames of depth map.

4. EXPERIMENT RESULTS

The experiment is implemented based on HTM3.0 [10] and evaluated according to the common test conditions (CTC) [11]. We code 100 frames in random access configure, intra period is set as 24, test sequences are from HEVC-3DV HD (1920x1088) test sequences. The anchor is HTM3.0, in which, the depth in loop filter is disabled. The bitrate includes all the bitrate of 3 view videos plus corresponding 3 depth map sequences; and the distortion is the average PSNR of 6 synthesized intermediate views.

4.1. Objective and subjective quality of synthesized views

Test objective quality results as shown in table 1, demonstrate that the average BD-Rate reduction is 0.65% in synthesized views in HD test sequence of HEVC-3DV, and the maximum BD-rate reduction is up to 1.6% for sequence UndoDancer. The proposed in loop filter has nearly no effect to the texture video; as the bitrate of depth only take a small proportion (about 5%-10%) of the total bitrate, so a small fluctuation of the depth bitrate is negligible to the total bitrate. The results reveal that more accurate the depth is, more efficient the proposed depth filter will achieve. We also evaluated the performance of the proposed depth in loop filter in comparison with the existing depth in loop filter methods in literature [4, 5 and 6]. As shown in table 2, our method is better.

Table **1** Objective quality improvement (BD-Rate results)

Seq (coding views)	Synthesized Views	Videos
GhostTownFly(1,5,9)	-0.9%	0.0%
PoznanHall2(5,6,7)	-0.3%	0.0%
PoznanStreet(3,4,5)	0.2%	0.0%
UndoDancer(1,5,9)	**-1.6%**	0.0%
Average	**-0.65%**	**0.0%**

Table **2** Different depth in loop filter methods comparison (BD-Rate results of synthesized views)

Depth filter methods	Proposed method	Method in [4]	Method in [5]	Method in [6]
BD-rate (%)	**-0.65**	9.08	-0.39	0.36

An example of quad-tree partition structure of TU in one depth frame is shown in fig 4. The characteristics of TU meet our previous hypothesis and the depth object edge is usually not coinciding with the TU boundary. So traditional in loop filter is not suitable for depth map.

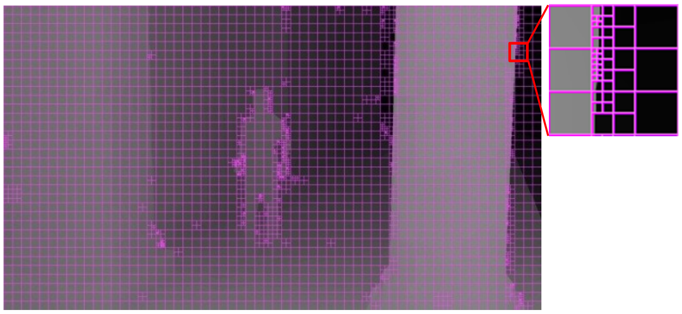


Fig 4. Quadtree partition structure of TU in depth map *UndoDancer_1920x1088* video view 5, frame 1, QP 35

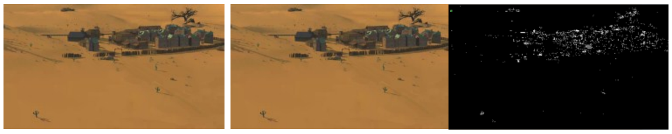

(a) (b) (c)

Fig 5. *GhostTownFly,* synthesized virtual view 8, frame1, QP 40. (a) Anchor (b) Proposal TU directional adaptive in loop filter (c) the difference (after binarization processing) between (a) and (b). The differences are mainly located at the edge of objects in depth map.

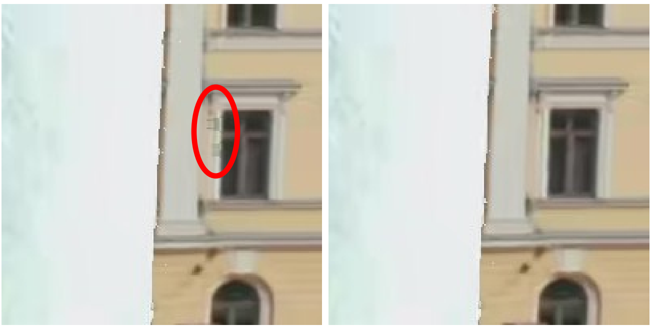

(a) (b)

Fig 6. Enlarge detail of the location in fig 4, *Undodancer,* synthesized virtual view 8, frame 13, QP 35. (a) Anchor; there is error in red elliptic ring area (b) Proposal TU directional adaptive in loop filter restores the flaw.

4.2. Complexity analysis

As the proposed depth in loop filter is only carried out in I frame of depth map that means coding 100 frames (including 100x3 texture frames plus 100x3 depth frames), and intra period is 24, only 5 depth frames in base view would be filtered. As the complexity of depth map coding is really high time consuming in HTM, the encoder and decoder time introduced by the proposed filter is negligible. Moreover all the TU could be filtered in parallel, so it could be accelerated with parallel processing.

5. CONCLUSION

In this paper, a content adaptive in-loop depth map filter is proposed for HEVC based 3DV video coding to preserve depth map details and improve the quality of the synthesized views. Experimental results demonstrate the effectiveness of our method. It can be integrated into HEVC-3DV depth map in loop filter.

ACKNOWLEDGEMENTS

This work was supported in part by Major State Basic Research Development Program of China (973 Program, 2009CB320903), National Science Foundation (61121002, 61103088, 61272255), National High-tech R&D Program of China(863 Program, SS2012AA010805), National Sci-Tech Support Plan(2011BAH08B01) .

REFERENCES

[1] H.Schwarz, K.Wegner, "Test Model under Consideration for HEVC based 3D video coding v3.0," ISO/IEC JTC1/SC29/WG11/N12744, Geneva, Switzerland, April 2012.

[2] P.List, A.Joch, J.Lainema, and G.Bjontegaard, etc. "Adaptive deblocking filter", IEEE TCSVT, vol.13, pp.614–619, 2003.

[3] B.Bross, W.Han, J.Ohm, and G.Sullivan, etc. "High Efficiency Video Coding (HEVC) text specification draft 8". ISO/IEC JTC1/SC29/WG11, JCTVC-J1003, Stockholm, SE, July 2012.

[4] F. Jäger, "3D-CE4.h Results on Trilateral Loop Filtering by RWTH Aachen University," ISO/IEC JTC1/SC29/WG11, M24870, Geneva, Switzerland, April 2012.

[5] I. Lim, H. Wey and J. Lee, "3D-CE4.h results on region based adaptive loop filter by Samsung," ISO/IEC JTC1/SC29/WG11, M24828, Geneva, Switzerland, April 2012.

[6] W. Lim, S. Yoo, and J. Nam, etc. "3D-HEVC-CE5 Region-based anisotropic filter for depth map," ISO/IEC JTC1/SC29/WG11, M23622, San Jose, USA, Feb 2012.

[7] Miska, M. Hannuksela, "Test Model for AVC-based 3D video coding v2.0," ISO/IEC JTC1/SC29/WG11, N12743, Geneva, Switzerland, July 2012.

[8] Z Xiong, X Sun, J Xu, and F Wu, "Content-adaptive deblocking for high efficiency video coding", Signal Processing: Image Communication, vol 27, issue 3, pp 260–268, March 2012.

[9] Y.Huang, C.Ming, C.Chen, etc. "In-Loop Adaptive Restoration," ISO/IEC JTC1/SC29/WG11, B077, Geneva, Switzerland, July 2010.

[10] HTM3.0 Reference Software, Available: https://hevc.hhi.fraunhofer.de/svn/svn_3DVCSoftware/tags/HTM-3.0/.

[11] "Common Test Conditions for 3DV experimentation," ISO/IEC JTC1/SC29/WG11, N12745, Geneva, Switzerland, April 2012.