|  |  |
| --- | --- |
| **Joint Collaborative Team on 3D Video Coding Extension Development**  **of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11**  1st Meeting: Stockholm, SE, 16–20 July 2012 | Document: JCT2-A0037 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **3D-CE3.a results on region based adaptive loop filter** | | |
| *Status:* | Input Document to JCT 3D | | |
| *Purpose:* | Proposal | | |
| *Author(s) or Contact(s):* | Ilsoon Lim  Hocheon Wey  Jaejoon Lee 14-1 Nongseo-dong Giheung-gu Yongin-si Gyeonggi-do 446-712 South Korea | Tel: Email: | +82 31 280 6688 ilsoon.lim@samsung.com |
| *Source:* | Samsung Electronics Co., Ltd. | | |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Abstract

In this proposal, we present a new depth map coding scheme to improve coding efficiency. For this purpose, we developed an adaptive range filter that restores the coded depth maps in accordance with the estimated noise variance. More specifically, each pixel is classified in such a way that the distortion cost is minimized and is restored depending on local noise level. The restored picture is used for the motion-compensated prediction of future pictures. This can improve coding efficiency. In particular, we propose to select the optimal fall-off parameter, which makes the range filter more powerful. Finally, depth coding that performs the new approach was implemented and this resulted in shows -0.60% (decoded view) and -1.70% (rendered view) BD-rate on average when compared with 3DV-ATM ver. 0.4 under the common test conditions.

# Introduction

Region-based adaptive loop filter (R-ALF) [2] for the depth coding is proposed and implemented based on 3DV-ATM 0.4. Experimental results shows -0.60% (decoded view) and -1.70% (rendered view) BD-rate for 3 view case, compared to anchor (3DV-ATM 0.4) and the decoding time has no more than 104.2%.

# Algorithm description

We develop an adaptive range filter [2][3][4] that restores the coded depth maps in accordance with the estimated noise variance at the in-loop structure. The restored depth map is used for the motion-compensated prediction of future pictures as shown in Fig 1 and Fig 2.



Fig. 1. The proposed encoder with R-ALF in depth coding



Fig. 2. The proposed decoder with R-ALF in depth coding

The range filter takes a weighted average of the pixels in a local neighbourhood; the weights depend on the intensity (photometric) distance. Mathematically, at a pixel location x, the output of the range filter is calculated as follows:

 (1)

where is a range variance controlling the weights in the intensity domain. is a spatial neighbourhood around x and K is the normalization constant.

Region-based adaptive loop filter (R-ALF) is based on the range filter. This is used in depth coding only. A frame is divided into two regions according to the estimated noise variance and the R-ALF is applied only to the high noisy region not to low noisy region. The reason is that the regional filtering can reduce the computational complexity in both the encoder and the decoder.

The adaptive range filter is performed with the associated fall-off parameter at the region. First, noise variance is estimated at every pixel of the coded frame. The approach is to measure the local intensity difference surrounding each pixel in the coded frame and use local intensity as the estimate for the pixel classification. In this document, we propose the estimated noise variance as follows:

(2)

where is the block containing the pixel x as shown in Fig. 3. The block-wise noise estimation can reduce the estimation time and make the algorithm faster.

After the noise estimation, we classify the each pixel with respect to the estimated noise variance as follows:

 (3)

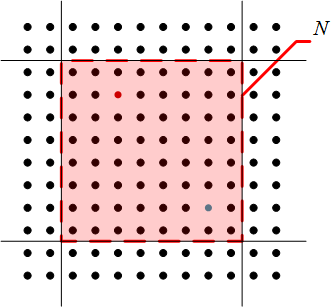


Fig. 3. Neighbourhood for the noise estimation for 8 × 8 block.

Fig. 4 shows an example of filtering map that indicates whether the filter is applied or not. The red square denotes the block that should be filtered. Note that the area of the filtered region depends on the threshold value. If we set the high threshold, only the small area will be set to filtered area. We fixed the threshold value () through all sequences.



Fig. 4. Illustration of the filtering map that indicates whether the range filter is applied or not.

For the high noisy region, we apply the range filter shown in Eq. (1). In the range filter, the choice of fall-off parameter () can have a significant effect on the coding efficiency. Unfortunately, there is not much theoretical basis on selecting the optimal fall-off parameter. In the video coding, it is of great advantage that we have access to the original picture and the coded picture at the encoder. This implies that we can determine the fall-off parameter, which makes the coded picture as close to the original picture as possible. The decoder can take advantage of such supplementary information determined and transmitted at the encoder.

The problem is to find the optimal fall-off parameter to minimize the distortion and can be formulated as

 (4)

If we assume that the distortion *D*(*σr*) is convex over every possible parameter, the above problem can be simplified to the evolution equation as follow:

 (5)

where is a parameter at iteration *t*, ∇ is the gradient of *D*, respectively. The above equation is computed as follows:

For =1,2,3, ..., t, ...

1. Compute *D*()

2. Compute the sign of *D*()- *D*(-1) when *σr*≥ 2

3. Repeat until the sign of *D*()- *D*(-1) is positive.

Finally,  is selected as the optimal fall-off parameter.

The advantage of this evolution equation is that, instead of the exhaustive search, we can make the quick search for the optimal parameter to minimize the depth distortion.

|  |  |
| --- | --- |
| C:\업무관련\논문발표\ICIP 2011\work\sigma_r_D_curve_QP27_FirstISlice_wide3.png | C:\업무관련\논문발표\ICIP 2011\work\sigma_r_D_curve_QP32_FirstISlice_wide3.png |
| 1. QPD=27 | 1. QPD=32 |

Fig. 5. Example of depth distortion curve when the different fall-off parameter () is selected.

A fall-off parameters per frame (range variance of high level region) are entropy-coded and transmitted to the decoder. The decoder also constructs a classification map by applying Eq. 1 to the decoded depth map. The restored depth map is stored in the Picture Buffer for Motion Estimation/Motion Compensation. It should be noted that the range filter does not necessarily lead to a better restoration performance. To account for this, one bit of side information should be transmitted per frame to signal to the decoder whether the restoration has to be used or not.

# Test results

## Coding performance

In this section, the objective coding performance of the proposed method is presented compared with 3DV-ATM 0.4 anchor EHP profile. The proposed filter is implemented based on 3DV-ATM 0.4 software. We followed the common testing condition document [5].

According to the common test condition, we turn off AVC-deblocking filter for depth map. We also report the test results in the appendix when we turn on AVC-deblocking filter compared to turn off the AVC-deblocking filter for depth.

Table 1 shows BD-rate gain in terms of PSNR of decoded views over texture + depth bit rate and BD-rate gain in terms of PSNR of synthesized views over texture + depth bit rate.

Table 1. Test results for the proposal in 3 view test scenario

|  |  |  |
| --- | --- | --- |
| Sequences | BD-rate with PSNR of  decoded views over  texture + depth bit rate | BD-rate with PSNR of  synthesized views over  texture + depth bit rate |
| Poznan Hall2 | -1.21 | -1.78 |
| Poznan Street | -0.17 | -1.08 |
| Undo Dancer | -1.19 | -3.18 |
| GT Fly | -0.36 | -1.31 |
| Kendo | -0.48 | -1.38 |
| Balloons | -0.52 | -1.32 |
| Newspaper | -0.27 | -1.86 |
| Average | -0.60 | -1.70 |
| Encoding Time (%) | 99.6% | |
| Decoding Time (%) | 104.2% | |

## Complexity performance

### Computing platform

|  |  |
| --- | --- |
| OS | : Microsoft Windows 7 Enterprise x64 Edition |
| PC | : Intel Xeon(R) CPU X5670@2.93Hz, 32GB RAM, 4TB Hard Disk |
| Compiler | : Microsoft Visual Studio C++ 2010 Professional(x64) |

### 3.2.2 Execution time measurements

The encoder software was single-threaded with no assembly code with no multi-core processing. The encoding times were collected from log files generated by the encoder, which were obtained using the same process as used by 3DV-ATM anchor.

The decoder software was single-threaded with no assembly code with no multi-core processing. The decoding times were collected from log files generated by the decoder, which were obtained using the same process as the anchor. As specified in the common testing condition document [5], the output YUV file generation are not included in the reported decoding times for both decoders. To disable the output YUV file generation, we set OutputRecYUVFile = 0 in decoder.cfg.

Table 1 shows encoding and decoding time ratio compared to the anchor. Our encoding time is similar to the anchor. Our decoder is only 104.2% of the anchor runtime compared to anchor.

### 3.2.3 Expected memory requirements

There is no memory footprint impact caused by region-based adaptive loop filter, because it merely uses the current decoded depth map and the current original depth map and replaces the decoded depth map with the restored depth map.

## AVC-deblocking filter vs. R-ALF

We have compared AVC-deblocking filter for depth with R-ALF as summarized in table 2. It is obvious that R-ALF gets the better of AVC-deblocking filter in all respects. In table 2, we set 3DV-ATM ver 0.4, which turns off the AVC-deblocking filter for depth as anchor (reference point). The single in-loop filtering is more beneficial in terms of the coding performance, the complexity performance, and the hardware design.

Table 2. AVC-deblocking filter vs. R-ALF in 3 view case

|  |  |  |  |
| --- | --- | --- | --- |
| Evaluation item | AVC-deblocking filter  for depth(A) | Region-based  adaptive loop filter(B) | Performance gap(B-A) |
| Dec. BD-rate | -0.44% | -0.60% | -0.16%***(better)*** |
| Syn. BD-rate | -0.72% | -1.70% | -0.98%***(better)*** |
| Decoding time | 102.4% | 104.2% | 1.8% |
| Encoding time | 100.0% | 99.6% | -0.4% |

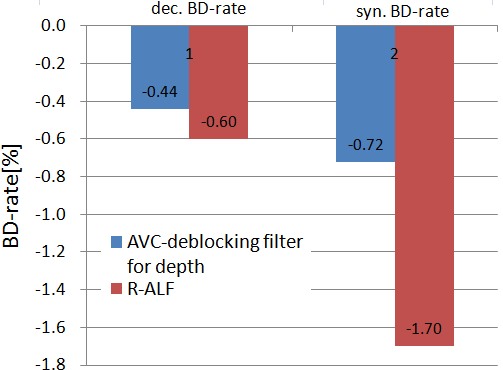


Fig. 6. Coding performance comparison between AVC-deblocking filter for depth and R-ALF.

# Syntax description

R-ALF syntax design is shown in Table 3. As it can be seen, two R-ALF parameters are transmitted in the slice header.

Table 3. Syntax related to R-ALF for slice header

|  |  |  |
| --- | --- | --- |
| slice\_header( ) { | C | Descriptor |
| … |  |  |
| **RALF\_param( )** |  |  |
| } |  |  |

|  |  |  |
| --- | --- | --- |
| **RALF\_param( ) {** | C | Descriptor |
| **RALF\_flag** |  | u(1) |
| **if( RALF\_flag ) {** |  |  |
| **RALF\_optimal\_parameter( )** |  | ue(v) |
| **}** |  |  |
| **}** |  |  |

**RALF\_flag** equal to 0 denotes that R-ALF is disabled for the current picture; equal to 1 denotes that ALF is enabled for the current picture.

**RALF\_optimal\_parameter** denotes the optimal variance of the range filter. This is integer ranging from 1 to 11 and transmitted once per slice.

# Conclusion

In this document, a region-based adaptive loop filter was proposed. Experimental results shows -0.60% (decoded view) and -1.70% (rendered view) BD-rate for 3 view case, compared to anchor and the decoding time has no more than 104.2%. We recommend that the proposal is integrated into 3DV-ATM

# Patent rights declaration(s)

**Samsung Electronics Co., Ltd. may have current or pending patent rights relating to the technology described in this contribution and, conditioned on reciprocity, is prepared to grant licenses under reasonable and non-discriminatory terms as necessary for implementation of the resulting ITU-T Recommendation | ISO/IEC International Standard (per box 2 of the ITU-T/ITU-R/ISO/IEC patent statement and licensing declaration form).**

# Appendix

## Performance evaluation for AVC-deblocking filter for depth

### Test conditions

|  |  |
| --- | --- |
| Anchor | : 3DV-ATM ver 0.4 |
|  | : We turn off AVC-deblocking filter for depth as stated in CTC. |
| Compared | : AVC-deblocking filter for depth On |

### Test results

Table 4. Test results for AVC-deblocking filter for depth in 3 view test scenario

|  |  |  |
| --- | --- | --- |
| Sequences | BD-rate with PSNR of  decoded views over  texture + depth bit rate | BD-rate with PSNR of  synthesized views over  texture + depth bit rate |
| Poznan Hall2 | -0.72 | -0.78 |
| Poznan Street | -0.16 | -0.54 |
| Undo Dancer | -0.76 | -0.36 |
| GT Fly | -0.22 | -0.62 |
| Kendo | -0.42 | -0.95 |
| Balloons | -0.50 | -0.93 |
| Newspaper | -0.27 | -0.89 |
| Average | -0.44 | -0.72 |
| Encoding Time (%) | 100.0% | |
| Decoding Time (%) | 102.4% | |

## Combination of R-ALF and post-processing filter

### Motivation

In last 100th MPEG meeting, the following two questions on R-ALF were raised.

* Question 1: R-ALF can be placed in the post structure
* Question 2: The synthesized coding gain of R-ALF can be obtained by other post filters.

In the subsection, we will give the reason and evidence why the above assumptions are not effective on R-ALF.

* Answer to question 1: In principle of video coding, the benefit of in-loop filter is to obtain the decoded coding gain. Using R-ALF, we have obtained -0.60% decoded coding gain in the in-loop structure and it is bigger than those of the built-in depth coding tools (e.g.: IVMP, AVC-deblocking filter for depth, DRWP).
* Answer to question 2: Someone does the post-processing on the decoded depth map for the depth enhancement such as Nokia’s nonlinear Depth Map Resampling (m24914) and Samsung’s dilation filter (JCT2-A0036). In this case, the synthesized coding gain of R-ALF is additive to such post-processing and even fortify the overall synthesized gain as shown in table 7. Thus, the synthesized coding gain of R-ALF can be additive to other post filter and cannot be covered with the post filter.

### Test result on combination of R-ALF and post-processing

In this test, we integrate R-ALF and dilation post filter to 3DV-ATM ver 0.4 to verify that the synthesized coding gain of R-ALF be additive to other post filter.

|  |  |
| --- | --- |
| Anchor | : 3DV-ATM ver 0.4 |
|  | : AVC-deblocking filter for depth Off |
| Compared | : The above anchor + R-ALF + dilation post filter(the reddish column below) |

Table 7. Combination of R-ALF and post-processing filter

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Sequences | R-ALF  (Table 1 in  JCT2-A0037) | | Dilation post filter  (Table 3.2 in  JCT2-A0036) | | R-ALF +  Dilation post filter  (Our integration test) | |
| BD-rate  (decoded) | BD-rate  (synthesized) | BD-rate  (decoded) | BD-rate  (synthesized) | BD-rate  (decoded) | BD-rate  (synthesized) |
| Poznan Hall2 | -1.21 | -1.78 | 0.00 | -0.11 | -1.21 | -2.06 |
| Poznan Street | -0.17 | -1.08 | 0.00 | -1.85 | -0.17 | -2.73 |
| Undo Dancer | -1.19 | -3.18 | 0.00 | -19.05 | -1.19 | -23.44 |
| GT Fly | -0.36 | -1.31 | 0.00 | -8.35 | -0.36 | -8.52 |
| Kendo | -0.48 | -1.38 | 0.00 | -0.31 | -0.48 | -1.82 |
| Balloons | -0.52 | -1.32 | 0.00 | 0.03 | -0.52 | -1.22 |
| Newspaper | -0.27 | -1.86 | 0.00 | -0.77 | -0.27 | -2.77 |
| Average | -0.60 | -1.70 | 0.00 | -4.35 | -0.60 | -6.08 |

# References

1. ISO/IEC JTC1/SC29/WG11“Description of AVC compatible 3D video coding technology by Samsung,” M22632, Nov. 2011.
2. ISO/IEC JTC1/SC29/WG11“3D-CE4.a results on region based adaptive loop filter by Samsung,” M24822, May. 2012.
3. I. Lim, H. Wey, and J. Lee, “Region-based Adaptive Bilateral Filter in Depth Map Coding,” Proc. IEEE International Conference on Image Processing, Sep. 2011.
4. I. Lim, and J. Lee, “Adaptive Nonlocal Range Filter in Depth Map Coding,” Proc. IEEE International Conference on Image Processing, Oct. 2012, to appear.
5. “Common Test Conditions for 3DV experimentation,” ISO/IEC JTC1/SC29/WG11/W12745, Geneva, Switzerland, May 2012.
6. ISO/IEC JTC1/SC29/WG11“Description of Nokia’s response to MPEG 3DV Call for Proposals on 3DV Video Coding Technologies,” M22552, Nov. 2011.
7. C. Tomasi, and R. Manduchi, “Bilateral Filtering for Gray and Color Images,” Proc. IEEE International Conference on Computer Vision, 1998.
8. I. Lim, H. Wey, J. Lee, and D. Park, “Trilateral Filter for Depth Map Interpolation in 3D Video,” in Society for Information Display (SID) Symposium Digest, June 2012.