|  |  |
| --- | --- |
| **Joint Collaborative Team on Video Coding (JCT-VC)**  **of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11**  4th Meeting: Daegu, KR, 20-28 January, 2011 | Document: JCTVC-D270\_r1 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **Low Complexity Parametric Adaptive Loop Filter** | | |
| *Status:* | Input Document to JCT-VC | | |
| *Purpose:* | Proposal | | |
| *Author(s) or Contact(s):* | E. Maani, W. Liu and L. Dong  1730 N 1st Street San Jose, CA 95112 | Tel:(408) 352-4710 Email:ehsan.maani@am.sony.com |  |
| *Source:* | Sony Electronics Inc. | | |

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

# Abstract

This document introduces a new type of Adaptive Loop Filter (ALF) intended to remove coding errors and improve compression efficiency. In this approach, a set of fixed filters is used instead of the traditional online-trained Weiner filters. The encoder chooses the best (in RD sense) filter for a Coding Unit (CU) and transmits the index of the filter in the bit stream to the decoder. The selection of the best filter at the encoder requires only a single-pass processing for each Large CU (LCU), therefore, reducing the encoder delay significantly compared to traditional ALF approaches which require one frame delay.

# Introduction

ALF is one of the most potent coding efficiency improvement tools introduced in recent years [1, 2, 3]. ALF can achieve coding gain by applying a set of filters (one or more) to various parts of the decoded picture in order to reduce compression noise and artifacts. Most existing ALF techniques use frame-level, online-trained, Wiener filters in conjunction with some type of block-based signaling to indicate to the decoder whether to apply filter a particular region of the image or not. The filter coefficients are typically different from one frame to another and thus need to be transmitted to the decoder as side information. Variable filter coefficients, however, makes the hardware implementation of these ALF techniques much more costly compared to fixed filters. Furthermore, the encoder optimization needed to find a “good” set of filters, is computationally expensive and, generally, requires multiple-pass processing of the images and their auto- and cross-correlation statistics. This results in an increase in the delay and memory bus bandwidth requirements of the encoder.

Parametric ALF (PALF), on the other hand, uses fixed filters and single-pass, LCU-level optimization to find the best filters at the encoder. In other words, once an LCU is reconstructed at the encoder and de-blocking filter is applied, the PALF process can start immediately to find an optimized set of filters for that LCU. No further processing for that LCU will be required once the filter indices are obtained. Therefore, the encoder can write the filter indices into the bit stream and start processing the next LCU as it becomes available at the encoder.

# Algorithm description

As the name suggests, parametric ALF consists of a set of fixed low-pass filters indentified by their parameters. In this document, we use direction and along-the-direction bandwidth as parameters, despite the fact that many other parameterization methods are also possible. Let be the impulse response of a filter with direction and bandwidth. The bandwidth determines the strength of the filter along the direction, while the filter is always close to all-pass in the direction perpendicular to , regardless of the value of . The values of and are discretized and the index of each value is transmitted in the bit-stream. It should be noted that the set of values of always includes an all-pass filter (no filtering). Figure 1 shows an example of for and four different values of .

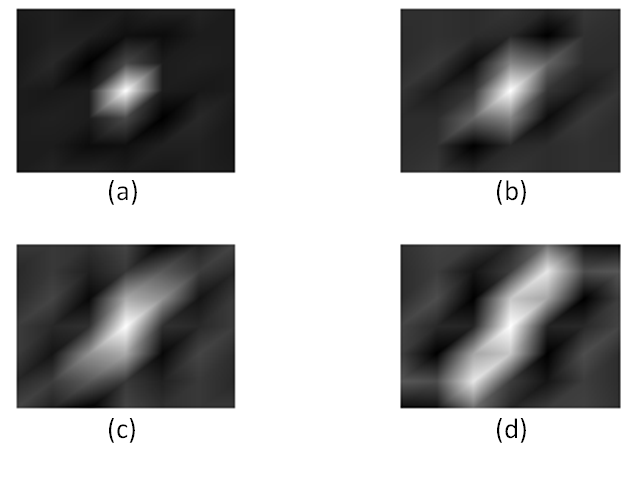


Figure 1. Impulse responses of filters with the same direction ( ) but different strengths. (a) Weak filter. (b) Medium weak filter. (c) Medium strong filter. (d) Strong filter.

At each LCU level, first, the maximum depth in the CU partition quadtree () -at which the PALF parameters are signaled- is transmitted to the decoder. Next, if the depth of a CU within the LCU is less than or equal to , both the filter strength level (corresponding to the value of ) and the direction index (corresponding to the value of ) are transmitted in the bit stream. Since the signaling method used for both parameters are similar, we only explain the signaling method used for the filter strength level . If the filter strength level is equal to the predicted strength level which is obtained from the left and top CUs, then only a 1 is transmitted. Otherwise, a 0 is transmitted and the actual level is signaled utilizing a fixed length code. Except for the case of the all-pass filter, signaling of the direction of the filter follows next.

The encoder optimization consists of finding the best max depth and the best filter indices (m, n) for all CUs within the LCU. This is done by finding the Sum of Squared Error (SSE) between the original block and the filtered version of the reconstructed block. The SSE associated with all possible filters should be computed in case of an exhaustive search method. However, one can envisage many fast search algorithms to avoid an exhaustive search. Starting from the first CU, in a Z-scan order, the filter with the minimum RD cost is selected for each CU. Then, the filter indices for that CU is recorded and the processing of the next CU begins. This process continues until the indices of the last CU is obtained. To further reduce the computational cost of the encoder, the signaling cost of each filter is estimated when an arithmetic entropy coding engine is used.

The two chroma channels use the same filter parameters; however, their filters are typically of smaller size compared to luma filters. Signaling of the filter parameters for the chroma channels is similar to that of luma, except that the maximum depth is typically of a smaller value. Both chroma components use the same filter strength and direction which minimizes the sum of their RD costs. For the filter direction of the chroma components, a flag is sent to signal whether the direction is the same as luma or not. If the direction is different, then, it is signaled similarly to the luma directions. Otherwise, no extra bits will be necessary and the directions of corresponding luma blocks are used for chroma components.

# Results

In our experiments, we assumed 8 possible directions and 5 possible filter strengths (including the all-pass filter). So, the total number of filters is 32 plus an all-pass filter. All 32 9x9 Wiener filters for luma and 32 5x5 Wiener filters for chroma are trained offline on a large set of video sequences. Exhaustive search is used to find the best filter indices. The algorithm is implemented in TMuC 0.9 (HM) and the results are obtained under the common test conditions [4] and compared to the HM anchors. Table 1, Table 2 and Table 3 show the results for the Intra only, low delay, and random access cases.

1. BD Rate for Intra Only

|  |  |  |  |
| --- | --- | --- | --- |
|  | Intra LoCo | | |
| Y BD-rate | U BD-rate | V BD-rate |
| Class A | -3.6 | -1.7 | -1.0 |
| Class B | -2.1 | -1.9 | -2.1 |
| Class C | -2.1 | -1.7 | -2.7 |
| Class D | -1.2 | -1.1 | -1.4 |
| Class E | -3.3 | -3.9 | -4.3 |
| All | -2.3 | -2.0 | -2.3 |

1. BD Rate for Random Access

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Random access LoCo |  |
| Y BD-rate | U BD-rate | V BD-rate |
| Class A | -2.6 | -2.3 | -1.5 |
| Class B | -3.4 | -6.4 | -5.8 |
| Class C | -2.2 | -5.2 | -6.6 |
| Class D | -1.6 | -4.1 | -4.8 |
| Class E |  |  |  |
| All | -2.5 | -4.9 | -5.2 |

1. BD Rate for Low Delay

|  |  |  |  |
| --- | --- | --- | --- |
|  | Low delay LoCo | | |
|  | Y BD-rate | U BD-rate | V BD-rate |
| Class A |  |  |  |
| Class B | -4.2 | -6.4 | -7.1 |
| Class C | -3.7 | -8.1 | -10.6 |
| Class D | -2.8 | -6.0 | -6.7 |
| Class E | -5.1 | -4.1 | -4.0 |
| All | -3.9 | -6.3 | -7.3 |

# References

1. T. Chujoh, A. Tanizawa and T. Yamakage, “Adaptive Loop Filter for Improving Coding Efficiency”, C402, Geneva, April 2008.
2. T. Chujoh, N. Wada and G. Yasuda, “Quadtree-based Adaptive Loop Filter,” ITU-T Q.6/SG16 Contribution, C181, Geneva, January 2009.
3. Marta Karczewicz, Peisong Chen, Rajan Joshi, Xianglin Wang, Wei-Jung Chien, Rahul Panchal, “Video coding technology proposal by Qualcomm Inc” JCTVC Contribution JCTVC-A121, Dresden, Germany, April 2010.
4. JCT-VC, Common test conditions and software reference configurations, JCTVC-C500, 3rd JCT-VC meeting, Guangzhou, October 2010.

# Patent rights declaration(s)

**Sony Corporation may have IPR 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).**