|  |  |
| --- | --- |
| **Joint Collaborative Team on Video Coding (JCT-VC)**  **of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11**  13th Meeting: Incheon, KR, 18–26 Apr. 2013 | Document: JCTVC-M0182 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **AHG17: complexity analysis of SHM1.0** | | |
| *Status:* | Input Document to JCT-VC | | |
| *Purpose:* | Information | | |
| *Author(s) or Contact(s):* | Jie Dong, Yan Ye, Yong He  9710 Scranton Rd, #250  San Diego, CA 92121 | Tel: Email: | +1-858-210-4809 {jie.dong,yan.ye,yong.he@interdigital.com} |
| *Source:* | InterDigital Communications Inc | | |

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

# Abstract

SHM1.0 complexity assessment was done as part of the AHG17 activities, and the anchor results were released in the JCTVC-L0440 package. Using the SHM1.0 anchor results in the JCTVC-L0440 package (labeled “r2”), this contribution summarizes the complexity of the PU-based RefIdx, the Picture-based RefIdx, and the IntraBL implementations. It is reported that the PU-based RefIdx and IntraBL solutions have similar complexity characteristics, whereas the complexity characteristics of the Picture-based RefIdx solution are very different. It is also reported that the RefIdx solution offers the design flexibility and allows different applications to choose from block-based or picture based implementation based on its specific complexity considerations.

# Introduction

At the 12th JCT-VC meeting, AHG17 was established to provide complexity assessment methodology to evaluate the complexity (memory bandwidth and computational complexity) of scalable coding tools. Since then, a group of companies and individuals worked together to provide the complexity assessment modules and SHM1.0 anchor results. These were released on Feb 23, and were used as anchors and reference modules to study the complexity of coding tools in SCE3 (combined prediction) and SCE4 (inter layer filtering). In this contribution, we analyze the anchor results provided by AHG17 to get a better understanding of the complexity of the two scalable solutions in SHM1.0. The results referred to in this contribution can be found in the latest JCTVC-L0440-v4 package [1].

SHM1.0 supports two scalable solutions, the RefIdx solution and the IntraBL solution. At the 12th JCT-VC meeting, the zero-motion restriction on the inter layer predicted PUs was adopted into the RefIdx framework [2] as a normative bitstream conformance requirement, because it was found that the zero-motion restriction provided not only complexity reduction, but also performance gain for the RefIdx solution. Because of this bitstream restriction, the RefIdx solution can be implemented in either PU-based approach or picture-based approach without the need to consider fractional-pixel interpolation due to non-zero inter-layer motion. For PU-based RefIdx implementation, the reconstructed BL blocks are upsampled on the fly and used in inter layer prediction; and the EL decoder is re-designed to support the on-the-fly upsampling. On the other hand, for picture-based RefIdx implementation, the entire BL reconstructed picture is upsampled and stored in the EL DPB, in a way that is transparent to the EL decoder and its block level operations. Because the IntraBL solution requires block level changes, only block-based implementation of IntraBL is considered.

The complexity results of SHM1.0 anchors released in the JCTVC-L0440-v4 package consist of the following three categories:

1. The block-based IntraBL implementation in JCTVC-L0440\_ComplexityAnalysis\_r2\_Anchor\_IntraBL.xls,
2. The PU-based RefIdx implementation in, JCTVC-L0440\_ComplexityAnalysis\_r2\_Anchor\_RefIdx\_PU\_based\_upsampling.xls, and
3. The picture-based RefIdx implementation in JCTVC-L0440\_ComplexityAnalysis\_r2\_Anchor\_RefIdx\_Pic\_based\_upsampling.xls.

Later in April, an updated JCTVC-L0440-v5 package was released, which additionally includes the complexity data for HM8.1 simulcast. The HM8.1 simulcast complexity data was used as the reference for complexity analysis below.

# Summary of complexity assessment results

Both average case complexity and worst case complexity are analyzed, as follows:

1. Average complexity: memory access and number of calculations due to upsampling (block based or picture based) and temporal motion compensation in the BL and EL are collected using the SHM1.0 anchor bitstreams coded under SHVC common test conditions [3].
2. Worst case complexity: for block-based implementation (PU-based RefIdx and IntraBL), memory access and number of calculations due to upsampling are computed assuming the smallest block size used in inter layer prediction; for picture-based implementation (Picture-based RefIdx), it is assumed that the entire BL picture is always upsampled using fixed block size, such as 16×16 blocks.

In SNR scalability, when upsampling is not performed, it is assumed that the BL reconstructed blocks (for block based implementations) or picture (for picture based implementation) are used directly.

## Average complexity

summarizes the average % complexity of IntraBL, PU-based RefIdx, and Picture-based RefIdx for RA, LDP and LDB. The anchor used to calculate the percentages in was HM8.1 simulcast, in which memory bandwidth and number of calculations consider temporal motion compensation in both BL and EL decoders. Note that only average numbers of the RA, LDP and LDB cases are included in , as the AI cases need to be treated differently for the reason explained below. Because blocks predicted using inter layer prediction (that is, upsampled base layer reconstruction) have no motion and therefore no need for fractional sample interpolation, there is net complexity reduction for IntraBL and PU-based RefIdx compared to the simulcast anchors. Additionally, it can be observed that IntraBL and PU-based RefIdx have very similar complexity (±1% difference). For picture-based RefIdx implementation, however, there is some complexity increase when compared to the simulcast anchors, especially in terms of # of calculations. A closer look at the accompanying xls files shows that the increase in complexity is higher when scaling ratio is higher (e.g., 2x compared to 1.5x has higher % increase); for SNR scalability, the Picture-based RefIdx also has a net complexity reduction (not reflected in ). This is because the picture-based RefIdx implementation upsamples the entire base layer reconstructed picture, regardless of whether certain blocks in the BL reconstruction are used in inter layer prediction or not.

Table 2‑1. Average % complexity compared to HM8.1 simulcast for test cases with motion compensated prediction in simulcast (including RA, LDP and LDB)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | | **Memory bandwidth** | | | **# calculations** | |
|  | | **8b/8b** | **64b/256b** | **64b/512b** | **Mults** | **Adds** |
| **IntraBL** | | 92% | 94% | 93% | 94% | 94% |
| **RefIdx** | **PU based** | 92% | 93% | 92% | 95% | 95% |
| **Pic based** | 103% | 101% | 101% | 144% | 144% |

For the AI test cases, the complexity data for HM8.1 simulcast are all zero, because there is no temporal motion compensation in BL and EL. Therefore, the simulcast data cannot be used as meaningful anchor, as it will only result in INF in all the cells in the table. For this reason, for the AI test cases, the PU-based and picture-based RefIdx are directly compared to IntraBL to get an understanding of relative complexity. Table 2‑2 summarizes the average % complexity of the RefIdx implementations using IntraBL as the reference. As can be seen, both PU-based RefIdx and Picture-based Refidx have lower complexity compared to IntraBL; in particular, the memory bandwidth for Picture-based RefIdx is only about half of that of IntraBL. This is because Picture-based RefIdx always upsamples the entire BL picture using fixed block size of 64x64, and has a fixed memory bandwidth overhead during upsampling. Therefore, its average complexity will be reduced when more EL blocks are inter layer predicted. In the AI cases, due to the lack of temporal prediction in the EL, the number of inter layer predicted blocks is significantly higher than the RA/LDP/LDB cases, which results in reduced average complexity for Picture-based RefIdx.

Table 2‑2. Average % complexity comparison between SHM1.0 RefIdx and SHM1.0 IntraBL for AI test cases

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Memory bandwidth** | | | **# calculations** | |
|  | **8b/8b** | **64b/256b** | **64b/512b** | **Mults** | **Adds** |
| **PU based RefIdx** | 91% | 89% | 89% | 98% | 98% |
| **Pic based RefIdx** | 56% | 42% | 43% | 98% | 98% |

## Worst case complexity

Another important measurement of complexity provided by AHG17 is the worst case complexity, which measures the peak complexity when certain prediction modes/PU sizes are used. Table 2‑3 compares the worst case complexity of HM8.1 with those of IntraBL, PU-based RefIdx, and Picture-based RefIdx in SHM1.0, for 2x and 1.5x spatial scalability. In terms of the worst case complexity, IntraBL and PU-based RefIdx also are similar, although the latter has higher peak memory bandwidth, because RefIdx allows PU-based inter layer prediction, whereas IntraBL allows only CU-based inter layer prediction. Picture-based RefIdx, on the other hand, has much lower worst case complexity. This is because Picture-based RefIdx always upsamples the entire BL picture, and hence its peak memory usage is not dependent upon the inter layer prediction block sizes (in other words there is no “peak”). For Picture-based RefIdx implementation, the numbers in JCTVC-L0440 shown in Table 2‑3 are calculated assuming that 16x16 fixed block units are fetched and upsampled; if larger block units are used in upsampling, e.g., 64x64 instead of 16x16, then the memory bandwidth overhead in Picture-based RefIdx will be further reduced.

SNR scalability is not shown in Table 2‑3 because SHM1.0 does not apply inter layer processing, and instead uses the reconstructed BL picture directly.

Table 2‑3. Worst case complexity analysis for IntraBL and RefIdx solutions in SHM1.0 (spatial scalability)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **HM8.1** | **IntraBL** | | **PU-based RefIdx** | | **Pic-based RefIdx** | |
| MC @ EL | Inter layer upsampling | % of HM8.1 | Inter layer upsampling | % of HM8.1 | Inter layer upsampling | % of HM8.1 |
| **Mult** | 74 | 37 | 50% | 37 | 50% | 24 | 32% |
| **Adds** | 63 | 32 | 51% | 32 | 51% | 21 | 33% |
| **MemBand(2D:4x2)** | 18 | 9 | 50% | 12 | 67% | 4 | 22% |
| **MemBand(2D:8x2)** | 27 | 14 | 52% | 18 | 67% | 5 | 19% |

# Conclusion

The IntraBL solution in SHM1.0 enables a block based implementation, whereas the RefIdx solution in SHM1.0 can be implemented either as block-based, or as picture-based. This contribution analyzes the complexity assessment anchor data provided by AHG17 in JCTVC-L0440. The following conclusions can be drawn from our analysis:

1. The block based implementations, based on either the RefIdx framework or the IntraBL framework, have similar complexity in both the average case and the worst case;
2. Compared to block based implementations, picture-based RefIdx implementation has very different complexity characteristics: for average case complexity, picture-based RefIdx has lower complexity for AI tests, but has higher complexity for RA/LDP/LDB tests; for worst case complexity, picture-based RefIdx has lower complexity.

Table 3‑1 summarizes our analysis by identifying the implementation that has lower complexity for a given case with an “X” in the corresponding cell. It can be seen that the RefIdx solution in SHM1.0 provides the design flexibility such that lower complexity can always be achieved depending on the application’s need.

Table 3‑1. Implementation category with lower complexity

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | **Average case complexity** | | **Worst case complexity** |
| AI | RA/LDP/LDB |
| **IntraBL** | |  | X |  |
| **RefIdx** | PU-based |  | X |  |
| Picture-based | X |  | X |

# References

1. E. François, A. Tabatabai, E. Alshina, “BoG report: Methodology for evaluating complexity of combined and residual prediction methods in SHVC”, JCTVC-L0440, January 2012.
2. X. Xiu, Y. He, Y. He, Y. Ye, “TE2: Results of test 3.2.1 on inter-layer reference picture placement”, JCTVC-L0051, January 2012.
3. X. Li, J. Boyce, P. Onno, Y. Ye, “Common SHM test conditions and software reference configurations”, JCTVC-L1009, January 2012.