|  |  |
| --- | --- |
| **Joint Collaborative Team on Video Coding (JCT-VC)**  **of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11**  19th Meeting: Strasbourg, FR, 17–24 Oct 2014 | Document: JCTVC-S0173 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **Non-CE6: Redundancy removal and simplification for Palette coding** | | |
| *Status:* | Input Document to JCT-VC | | |
| *Purpose:* | Proposal | | |
| *Author(s) or Contact(s):* | Yuwen He, Xiaoyu Xiu, Yan Ye 9710 Scranton Rd, #250 San Diego, CA 92121, USA | Tel: Email: | +1-858-210-4819 [yuwen.he@interdigital.com](mailto:yuwen.he@interdigital.com) [xiaoyu.xiu@interdigital.com](mailto:xiaoyu.xiu@interdigital.com) [yan.ye@interdigital.com](mailto:yan.ye@interdigital.com) |
| *Source:* | InterDigital Communications, Inc. | | |

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

# Abstract

This proposal proposes to remove some syntax signalling redundancy and simplify the binarization method of palette\_run value for palette coding. Compared to CE6 anchors, the lossy coding with proposed technologies achieves {Y, U, V} BD rate gain of {-0.2%, -0.2%, -0.2%}, {-0.2%, -0.2%, -0.2%} and {-0.2%, -0.2%, -0.3%} for the category (RGB/YUV, text & graphics with motion, 1080p and 720p) for AI, RA and LDB, respectively. For lossless coding,it achieves bitratre saving of 0.1%, 0.1% and 0.1% for the category (RGB, text & graphics with motion, 1080p) for AI, RA and LDB, respectively.

# Introduction

The proposal proposes two modifications for palette coding.

The first modification is to add condition checking for syntax elements palette\_escape\_val\_present\_flag, palette\_mode and palette\_run to remove signaling redundancy. palette\_escape\_val\_present\_flag will not be signaled when palette table is empty. palette\_mode and palette\_run will not be signalled when there is only one color in the CU due to one of two possibilities: 1) there is only one color in the palette table and there is no escape color, or 2) the palette table is empty.

The syntax changes for palette coding is highlighted.

|  |  |
| --- | --- |
| Palette\_coding( x0, y0, nCbS ) { | Descriptor |
| **palette\_transpose\_flag** | ae(v) |
| **palette\_share\_flag**[ x0 ][ y0 ] | ae(v) |
| … |  |
| if(palette\_size >0) |  |
| **palette\_escape\_val\_present\_flag** | ae(v) |
| if( palette\_escape\_val\_present\_flag ) |  |
| indexMax = palette\_size |  |
| else |  |
| indexMax = palette\_size − 1 |  |
| scanPos = 0 |  |
| while( scanPos < nCbS \* nCbS ) { |  |
| xC = x0 + travScan[ scanPos ][ 0 ] |  |
| yC = y0 + travScan[ scanPos ][ 1 ] |  |
| if( scanPos > 0) { |  |
| xC\_prev = x0 + travScan[ scanPos − 1 ][ 0 ] |  |
| yC\_prev = y0 + travScan[ scanPos − 1 ][ 1 ] |  |
| } |  |
| if(indexMax>0 && scanPos > = nCbS && palette\_mode[xC\_prev][yC\_prev] ! = COPY\_ABOVE ) |  |
| **palette\_mode**[ xC ][ yC ] | ae(v) |
| if( palette\_mode[ xC ][ yC ] ! = COPY\_ABOVE ) { |  |
| adjustedIndexMax = indexMax |  |
| adjustedRefIndex = indexMax + 1 |  |
| } |  |
| if( scanPos > 0 && palette\_mode[xC\_prev][yC\_prev] ! = ESCAPE ) { |  |
| if( palette\_mode[xC\_prev][yC\_prev] = = INDEX ) { |  |
| adjustedIndexMax − = 1 |  |
| adjustedRefIndex = paletteMap[ xC\_prev ][ yC\_prev ] |  |
| } |  |
| if( scanPos > = nCbS && palette\_mode[ xC\_prev ][ yC\_prev ] = = COPY\_ABOVE   && palette\_mode[ xC ][ yC − 1 ] ! = ESCAPE ) { |  |
| adjustedIndexMax − = 1 |  |
| adjustedRefIndex = paletteMap[ xC ][ yC − 1 ] |  |
| } |  |
| } |  |
| if(palette\_mode[ xC ][ yC ] ! = COPY\_ABOVE ) { |  |
| if( adjustedIndexMax > 0 ) |  |
| **palette\_index** | ae(v) |
| if( palette\_index > = adjustedRefIndex ) |  |
| palette\_index++ |  |
| if( palette\_index = = palette\_size ) { |  |
| for( cIdx = 0; cIdx < 3; cIdx++ ) { |  |
| **palette\_escape\_val** | ae(v) |
| paletteEscapeVal[ cIdx ][ xC ][ yC ] = palette\_escape\_val |  |
| } |  |
| palette\_mode[ xC ][ yC ] = ESCAPE |  |
| scanPos++ |  |
| } |  |
| } |  |
| if( indexMax>0 && palette\_mode[xC][yC] ! = ESCAPE ) { |  |
| **palette\_run** | ae(v) |
| runPos = 0 |  |
| runMode = palette\_mode[ xC ][ yC ] |  |
| while ( runPos < = palette\_run ) { |  |
| xC = x0 + travScan[ scanPos ][ 0 ] |  |
| yC = y0 + travScan[ scanPos ][ 1 ] |  |
| if( palette\_mode[ xC ][ yC ] = = INDEX ) { |  |
| palette\_mode[ xC ][ yC ] = INDEX |  |
| paletteMap[ xC ][ yC ] = palette\_index |  |
| } else { |  |
| palette\_mode[ xC ][ yC ] = COPY\_ABOVE |  |
| paletteMap[ xC ][ yC ] = paletteMap[ xC ][ y − 1 ] |  |
| } |  |
| runPos++ |  |
| scanPos++ |  |
| } |  |
| } |  |
| } |  |
| … |  |
| } |  |

The second modification is to simplify the binarization method for syntax element palette\_run. In SCM-2.0 design, the whole value range of palette\_run is partitioned into three segments and each segment will use a different binarization method. As shown in Table 1, in the second segment, Golomb-Rice code is used to code the suffix, and in the third segment, Exp-Golomb code is used to code the suffix. We propose to combine the second and the third segments and only use Exp-Colomb code to encode the suffix.

Table . Binarization of palette\_run value in SCC [2]

|  |  |  |
| --- | --- | --- |
| Run | Bin |  |
| 0 | 0 | Segment1: greater\_than\_0\_flag, greater\_than\_1\_flag, greater\_than\_2\_flag |
| 1 | 10 |
| 2 | 110 |
| 3 | 111 0000 | Segment2: prefix “111” + suffix Golomb-Rice code with rice parameter equal to 3 |
| … | 111 0xxx |
| 11 | 111 10000 |
| … | 111 10xxx |
| 19 | 111 110000 |
| … | 111 110xxx |
| 27 | 111 1110000 | Segment3: prefix “111” + suffix Exp-Golomb code starting from codeNum 7 in Table 9-1 [2] |
| … | 111 1110xxx |
| 35 | 111 111100000 |
| … | 111 11110xxxx |

Table . Proposed binarization of palette\_run value

|  |  |  |
| --- | --- | --- |
| Run | Bin |  |
| 0 | 0 | Segment1: greater\_than\_0\_flag, greater\_than\_1\_flag, greater\_than\_2\_flag |
| 1 | 10 |
| 2 | 110 |
| 3 | 111 000 | Segment2: prefix “111” + 2nd order Exp-Golomb code |
| … | 111 0xx |
| 7 | 111 10000 |
| … | 111 10xxx |
| 15 | 111 1100000 |
| … | 111 110xxxx |
| 31 | 111 111000000 |
| … | 111 1110xxxxx |
| 63 | 111 11110000000 |
| … | 111 11110xxxxxx |

# Simulation results

The compression performance is measured using BD rate compared with CE6 anchors, using the CE6 test conditions [1]. Table 3 and Table 4 give the detailed average BD rate reduction for lossy and lossless coding with proposed redundancy removal and simplification compared with CE6 anchors, respectively. The full test results are provided with the accompanying spreadsheets. The separate test results are also included.

SCM2.0-vs-PLTS\_SingleColor\_lossless\_g8.xlsm: lossless coding results for first modification only.

SCM2.0-vs-PLTS\_SingleColor\_lossy\_g8.xlsm: lossy coding results for first modification only.

SCM2.0-vs-PLTS\_RunCoding\_lossless\_g8.xlsm: lossless coding results for second modification only.

SCM2.0-vs-PLTS\_RunCoding\_lossy\_g8.xlsm: lossy coding results for second modification only.

SCM2.0-vs-PLTSimplify\_lossless\_g8.xlsm: lossless coding results for two modifications.

SCM2.0-vs-PLTSimplify\_lossy\_g8.xlsm: lossy coding results for two modifications.

As shown in Table 3, compared with CE6 anchors, the lossy coding achieves average {Y, U, V} BD rate gain of {-0.2%, -0.2%, -0.2%}, {-0.2%, -0.2%, -0.2%} and {-0.2%, -0.2%, -0.3%} for the category (RGB/YUV, text & graphics with motion, 1080p and 720p) for AI, RA and LDB, respectively.

Table 3. Average BD rate reduction for lossly coding compared with CE6 anchors

|  |  |  |  |
| --- | --- | --- | --- |
|  | **All Intra** | | |
|  | G/Y | B/U | R/V |
| RGB, text & graphics with motion, 1080p | -0.2% | -0.2% | -0.2% |
| RGB, text & graphics with motion,720p | -0.2% | -0.2% | -0.2% |
| RGB, mixed content, 1440p | -0.1% | -0.1% | -0.1% |
| RGB, mixed content, 1080p | -0.1% | -0.1% | -0.1% |
| RGB, Animation, 720p | 0.0% | 0.0% | 0.0% |
| RGB, camera captured, 1080p | 0.0% | 0.0% | 0.0% |
| YUV, text & graphics with motion, 1080p | -0.3% | -0.3% | -0.3% |
| YUV, text & graphics with motion,720p | -0.2% | -0.2% | -0.2% |
| YUV, mixed content, 1440p | -0.1% | -0.1% | -0.2% |
| YUV, mixed content, 1080p | -0.1% | -0.1% | -0.2% |
| YUV, Animation, 720p | 0.0% | 0.0% | 0.0% |
| YUV, camera captured, 1080p | 0.0% | 0.0% | 0.0% |
| Enc Time[%] | 102% | | |
| Dec Time[%] | 101% | | |
|  |  |  |  |
|  | **Random Access** | | |
|  | G/Y | B/U | R/V |
| RGB, text & graphics with motion, 1080p | -0.1% | -0.1% | -0.1% |
| RGB, text & graphics with motion,720p | -0.3% | -0.3% | -0.3% |
| RGB, mixed content, 1440p | 0.0% | -0.1% | -0.1% |
| RGB, mixed content, 1080p | 0.0% | 0.0% | -0.2% |
| RGB, Animation, 720p | 0.0% | 0.0% | 0.0% |
| RGB, camera captured, 1080p | 0.0% | 0.0% | 0.0% |
| YUV, text & graphics with motion, 1080p | -0.1% | -0.1% | -0.1% |
| YUV, text & graphics with motion,720p | -0.3% | -0.2% | -0.3% |
| YUV, mixed content, 1440p | -0.1% | -0.2% | -0.3% |
| YUV, mixed content, 1080p | 0.0% | -0.2% | -0.2% |
| YUV, Animation, 720p | 0.0% | 0.0% | 0.0% |
| YUV, camera captured, 1080p | 0.0% | 0.1% | 0.1% |
| Enc Time[%] | 102% | | |
| Dec Time[%] | 101% | | |
|  |  |  |  |
|  | **Low delay B** | | |
|  | G/Y | B/U | R/V |
| RGB, text & graphics with motion, 1080p | -0.2% | -0.2% | -0.2% |
| RGB, text & graphics with motion,720p | -0.2% | -0.3% | -0.2% |
| RGB, mixed content, 1440p | 0.1% | 0.1% | 0.0% |
| RGB, mixed content, 1080p | -0.1% | 0.5% | 0.1% |
| RGB, Animation, 720p | 0.0% | 0.0% | 0.0% |
| RGB, camera captured, 1080p | 0.0% | 0.0% | 0.0% |
| YUV, text & graphics with motion, 1080p | -0.2% | -0.2% | -0.2% |
| YUV, text & graphics with motion,720p | -0.3% | -0.2% | -0.5% |
| YUV, mixed content, 1440p | -0.1% | -0.2% | -0.1% |
| YUV, mixed content, 1080p | 0.1% | 0.5% | 0.0% |
| YUV, Animation, 720p | 0.0% | -0.3% | -0.1% |
| YUV, camera captured, 1080p | 0.0% | 0.0% | 0.1% |
| Enc Time[%] | 102% | | |
| Dec Time[%] | 100% | | |

Table 4. Average BD rate reduction for lossless coding compared with CE6 anchors

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **All Intra** | | | |
|  | Bit-rate saving (Total) | Bit-rate saving (Average) | Bit-rate saving (Min) | Bit-rate saving (Max) |
|  |
| RGB, text & graphics with motion, 1080p | 0.1% | 0.1% | 0.1% | 0.1% |
| RGB, text & graphics with motion,720p | 0.1% | 0.1% | 0.0% | 0.1% |
| RGB, mixed content, 1440p | 0.0% | 0.0% | 0.0% | 0.0% |
| RGB, mixed content, 1080p | 0.0% | 0.0% | 0.0% | 0.0% |
| RGB, Animation, 720p | 0.0% | 0.0% | 0.0% | 0.0% |
| RGB, camera captured, 1080p | 0.0% | 0.0% | 0.0% | 0.0% |
| YUV, text & graphics with motion, 1080p | 0.1% | 0.1% | 0.0% | 0.2% |
| YUV, text & graphics with motion,720p | 0.1% | 0.1% | 0.0% | 0.1% |
| YUV, mixed content, 1440p | 0.0% | 0.0% | 0.0% | 0.0% |
| YUV, mixed content, 1080p | 0.0% | 0.0% | 0.0% | 0.0% |
| YUV, Animation, 720p | 0.0% | 0.0% | 0.0% | 0.0% |
| YUV, camera captured, 1080p | 0.0% | 0.0% | 0.0% | 0.0% |
| Enc Time[%] | 102% | | | |
| Dec Time[%] | 102% | | | |
|  |  |  |  |  |
|  | **Random Access** | | | |
|  | Bit-rate saving (Total) | Bit-rate saving (Average) | Bit-rate saving (Min) | Bit-rate saving (Max) |
|  |
| RGB, text & graphics with motion, 1080p | 0.1% | 0.1% | 0.1% | 0.2% |
| RGB, text & graphics with motion,720p | 0.0% | 0.1% | 0.0% | 0.1% |
| RGB, mixed content, 1440p | 0.0% | 0.0% | 0.0% | 0.0% |
| RGB, mixed content, 1080p | 0.0% | 0.0% | 0.0% | 0.0% |
| RGB, Animation, 720p | 0.0% | 0.0% | 0.0% | 0.0% |
| RGB, camera captured, 1080p | 0.0% | 0.0% | 0.0% | 0.0% |
| YUV, text & graphics with motion, 1080p | 0.1% | 0.1% | 0.1% | 0.2% |
| YUV, text & graphics with motion,720p | 0.0% | 0.1% | 0.0% | 0.1% |
| YUV, mixed content, 1440p | 0.0% | 0.0% | 0.0% | 0.0% |
| YUV, mixed content, 1080p | 0.0% | 0.0% | 0.0% | 0.0% |
| YUV, Animation, 720p | 0.0% | 0.0% | 0.0% | 0.0% |
| YUV, camera captured, 1080p | 0.0% | 0.0% | 0.0% | 0.0% |
| Enc Time[%] | 102% | | | |
| Dec Time[%] | 102% | | | |
|  |  |  |  |  |
|  | **Low Delay B** | | | |
|  | Bit-rate saving (Total) | Bit-rate saving (Average) | Bit-rate saving (Min) | Bit-rate saving (Max) |
|  |
| RGB, text & graphics with motion, 1080p | 0.1% | 0.1% | 0.0% | 0.2% |
| RGB, text & graphics with motion,720p | 0.0% | 0.0% | 0.0% | 0.1% |
| RGB, mixed content, 1440p | 0.0% | 0.0% | 0.0% | 0.0% |
| RGB, mixed content, 1080p | 0.0% | 0.0% | 0.0% | 0.0% |
| RGB, Animation, 720p | 0.0% | 0.0% | 0.0% | 0.0% |
| RGB, camera captured, 1080p | 0.0% | 0.0% | 0.0% | 0.0% |
| YUV, text & graphics with motion, 1080p | 0.0% | 0.1% | 0.0% | 0.1% |
| YUV, text & graphics with motion,720p | 0.0% | 0.0% | 0.0% | 0.1% |
| YUV, mixed content, 1440p | 0.0% | 0.0% | 0.0% | 0.0% |
| YUV, mixed content, 1080p | 0.0% | 0.0% | 0.0% | 0.0% |
| YUV, Animation, 720p | 0.0% | 0.0% | 0.0% | 0.0% |
| YUV, camera captured, 1080p | 0.0% | 0.0% | 0.0% | 0.0% |
| Enc Time[%] | 103% | | | |
| Dec Time[%] | 103% | | | |

# Patent rights declaration(s)

**InterDigital Communications, Inc. 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).**

# References

1. Y.-W. Huang, P. Onno, R. Cohen, V. Seregin, X. Xiu, Z. Ma, “Description of Core Experiment 6 (CE6): Improvements of palette mode”, JCTVC-R1106, Jul. 2014, Sapporo, JP.
2. R. Joshi, J. Xu, “HEVC Screen Content Coding Draft Text 1”, JCTVC-R1005, Jul. 2014, Sapporo, JP.