

JVT-T086

Adaptive Variable Length Coding for FGS

Yan Ye, Yiliang Bao
Qualcomm Inc.

Review of the significance coding in JD6

- Send significant coefficient positions using zero runs and EOB
 - EOB is assigned a symbol index, I_{EOB} , referred to as EOB Shift
 - The index of a zero run R , $I_R = (R < I_{EOB}) ? R : (R + 1)$
- Index of a symbol is coded using VLC from 1 of 5 codebooks
- Both EOB shift and codebook selection depend on coding cycle
- Optimal tables are searched and sent before the slice is coded
- Signal whether a block has coefficients of magnitude larger than 1
 - Token "*largeCoeffToken*" is formed to indicate
 - » Number of large coefficients (magnitude > 1), and the largest magnitude
 - Offset " $L + 1$ ", in which " L " is the number of coefficients after last significant coefficients
 - » " $L + 1$ " to accommodate all possible runs and regular EOB
 - Send the symbol " $L + 1 + \text{largeCoeffToken}$ " using selected cookbook

Issues with the current design

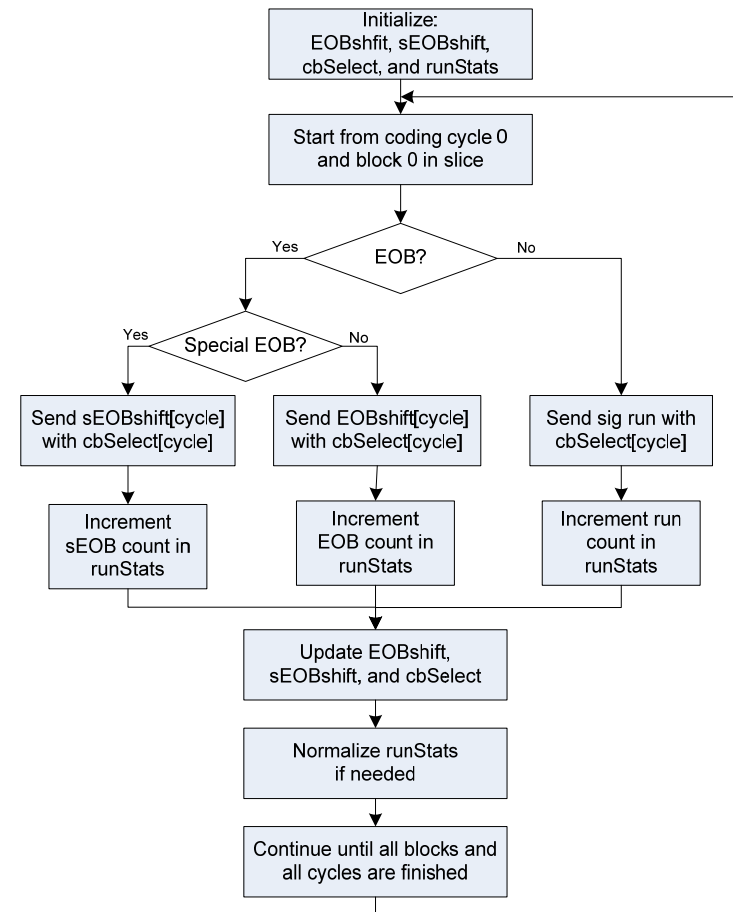
- Perform two-pass encoding on significant coefficients
 - First pass to find the optimal coding parameters
 - Apply coding parameters to the actual encoding in the second pass
 - Two-pass encoding entails high implementation complexity, will be a challenge for many applications
- Special EOB can be more probable than that implied by “ $L + 1$ ”
- Coding of chroma DC/AC is not tuned toward their statistics
 - Same EOB shift table for Chroma DC and AC
 - Same codebook selection for luma, chroma DC and chroma AC
- 8x8 coding uses only unary codes

What are proposed

- Add special EOB shift table
 - Use an offset dynamically adjustable to match the actual probability, instead of “ $L + 1$ ”
 - *largeCoeffToken* not changed
- Determine the optimal coding parameters on the fly, based on symbol statistics
 - Enable one-pass encoding
- Use separate coding parameters for luma and chroma
- Use 8x8 to 4x4 de-interleaving in AVC for 8x8 coding

The adaptive VLC system

- Maintain counts of zero runs, EOB and special EOB
- Update the coding tables based on accumulated statistics
- Table update and statistics normalization happen at MB boundary



The accumulated statistics

- To reduce complexity, only following statistics is maintained
 - Cycle 0, counts of run=0, 1, 2, and total count of all other runs
 - Cycle 1 to 9, counts of EOB, sEOB, run = 0, 1, 2, and total count of all other runs
 - Cycle 10 to 15, no statistics for these cycles are maintained
- Normalization of statistics
 - For each coding cycle, the probability counts are scaled down by a factor of 2 when the MPS count reaches 128

Update of coding parameters

- Low complexity algorithms for selecting the optimal coding parameters
- EOB shift and special EOB (sEOB) shift
 - Probability of EOB and sEOB are compared with those of runs $R = 0, 1, 2$, to determine the optimal symbol index
 - sEOB always takes a larger symbol than EOB
- Optimal coding table selection based on 4 variables
 - Counts of 3 most probable symbols, A, B, C, &
 - Count of all other symbols, D
- Remove CB3 in JD6 to reduce complexity

Improvement over JSVM_5_3, FReXt on

- Presented as percentage of overall bit rate reduction
- SNR test conditions (base QP/GOP size), 3 FGS layers
- JSVM_5_3 has all FGS VLC proposals accepted, with CBP fix

Sequence	QCIF 15FPS	CIF 30FPS	4CIF 60FPS
Bus	-5.80	-8.50	
Football	-3.73	-6.92	
Foreman	-6.28	-9.76	
Mobile	-3.65	-7.00	
City	-5.85	-9.63	-14.25
Crew	-5.48	-7.11	-11.34
Harbour	-6.23	-9.15	-12.54
Soccer	-4.85	-8.56	-14.16
Average	-5.23	-8.33	-13.07

Additional performance comparison

- JSVM_5_3, FRext on, 8x8 blocks are coded as 4x4 blocks using de-interleaving scheme defined in AVC
 - T086 achieves a reduction of overall bit rate up to 3.7%, for 1 FGS layer
- JSVM_5_3, FRext off
 - T086 achieves a reduction of overall bit rate up to 2.7%, for 1 FGS layer

Summary

- Propose 1-pass encoding for FGS VLC coder
 - Low complexity algorithms for selecting the optimal coding parameters
- Achieve significant coding gains over following 3 cases
 - JSVM_5_3, FRext off
 - JSVM_5_3, FRext on
 - JSVM_5_3, FRext on, but 8x8 block coded as 4x4 blocks as in AVC
- Propose to adopt the modifications to JD