Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG
(ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6)
16th Meeting: Poznań, PL, 24-29 July, 2005

Document: JVT-P056
Filename: JVT-P056.doc

Title:	**Variable length codes for SVC**
Status:	Input Document to JVT
Purpose:	Proposal

Author(s) or Contact(s):
Justin Ridge / Marta Karczewicz
6000 Connection Dr., MS 3-4-1400
Irving, TX 75039
USA
Tel: +1 972 374 0628
Email: justin.ridge @ nokia.com

Source: Nokia

Abstract

As noted at the previous JVT meeting, context-adaptive variable-length codes (CAVLC) is a feature currently missing from the JSVM software. Nokia has rectified this by incorporating CAVLC into the JSVM, and verified base-layer compatibility with H.264/AVC by decoding test sequences using H.264/AVC JM 9.6. For enhancement layers, the performance was found to be in line with the single-layer case.

Introduction

Non-scalable H.264/AVC provides a choice of two entropy coders, CABAC and CAVLC. It is generally accepted that CABAC offers superior coding efficiency at the price of complexity. CAVLC is particularly appealing for use in complexity-constrained devices, such as mobile terminals.

If the scalable extension to H.264/AVC is to be useful in a broad range of scenarios, it should be capable performing adequately on complexity-constrained devices. To this end, we believe that CAVLC is necessary, at least for the base layer and quality enhancements.

In the case of quality enhancement layers, CAVLC offers the further advantage that it simplifies the truncation procedure, since there is no internal buffer. Conversely, the current enhancement layer CABAC decoder must use exception handling in order to recover from buffer underflow situations.

Nokia has implemented CAVLC in JSVM 1.2.5, and proposes committing these modifications to the CVS server along with other “missing features” following the meeting. As with the H.264/AVC JM software, the switch between CABAC and CAVLC would be an encoder configuration option.

Modifications proposed

The existing JSVM software requires almost no structural modification for incorporation of CAVLC. In fact, a CAVLC encoder is partially implemented in order to generate bit counts for R/D optimization.

We have essentially completed four tasks:

1. Implementation of missing CAVLC encoder functions
2. Addition of a UvlcReader class for CAVLC decoding

3. Extension of baseline CAVLC to handle main profile encoding, for example quarter-pel motion vector differentials.
4. Conversion of FGS-specific CABAC functions to CAVLC.

The first three are more or less routine programming tasks and do not merit a detailed discussion. Interested parties should view the accompanying source code for details. The remainder of this section describes how CAVLC encodes quality enhancement (FGS) data.

Overview of CAVLC for FGS

In the significance pass, rather than coding each coefficient individually, a single VLC symbol is emitted for each cycle in cyclical block coding, specifying the number of coefficients processed. According to cyclical block coding, all but the last of these coefficients must be zero, hence the structure is already well-suited to a "run-level" approach without modification.

Encoding the value zero (i.e. no coefficients processed) indicates an end-of-block (EOB). A sign flag follows the terminating value, but magnitude is initially assumed to be unitary (no magnitude value is encoded).

After the EOB marker, the terminating values are grouped and run-level codes are transmitted to indicate the position and exact magnitude of those with magnitude greater than one.

In the refinement pass, the refinement bits are statistically more likely to be zero. Consequently the refinement bits in a given 4x4 or 8x8 block are grouped into fours, and a single VLC codeword is generated for each.

The concept of using a "coded block pattern" (CBP) is borrowed from regular H.264/AVC. We do not use a context, and omit the bits from blocks already coded in the base layer, but otherwise the mechanism is unchanged.

Context selection

All VLCs used for encoding coefficients in the significance pass are modified start-step-stop codes, as previously described in JVT-C086 and VCEG-L19. These codes are characterized by a cutoff parameter, 'm'. Symbols less than or equal to 'm' are encoded using an exponential Golomb (expGolomb) code. A symbol 'C' with value greater than 'm' is encoded in two parts: a prefix of ones, length

$$p = \left\lfloor \frac{C-m}{3} \right\rfloor + m$$

and a "remainder" suffix of either 00, 01, or 10. The codewords corresponding to m=2 and m=4 are shown in Table 1.

Symbol	m=1	m=5
0	1: 0	1: 0
1	3: 100	2: 10
2	3: 101	3: 110
3	3: 110	4: 1110
4	5: 11100	5: 11110
5	5: 11101	7: 1111100
6	5: 11110	7: 1111101
7	7: 1111100	7: 1111110

Table 1: VLCs for m=1, m=5

The probability of an EOB is greatly influenced by cycle number. A vector specifying the EOB offsets to be used for each cycle is encoded once per frame. This vector is constrained to be monotonically decreasing as cycle number increases. For example, the vector {4 2 1 1 0 0 …} means that the EOB should be symbol 4 in cycle 0, symbol 2 in cycle 1, symbol 1 in cycles 2 &

3, and symbol 0 (most probable symbol) in subsequent cycles. The first value (4) is encoded using a Golomb code with k=2, followed by the number of times each offset appears in the vector. For example, 4 appears no more times after the first value, so 0 is encoded; 3 does not appear at all, so 0 is encoded; 2 appears once so 1 is coded; 1 appears twice so 2 is coded. Thus the symbols coded are {0,0,1,2}. Performing this coding once per frame incurs minimal overhead.

A table mapping the cycle number and position of the last-encoded coefficient in the base layer (LEBL) to the VLC characteristic 'm' is encoded once per frame. The optimal mapping varies significantly not only from one sequence to another, but also between frames within a given sequence.

Because the base layer must be processed before encoding the enhancement layer, analyzing the base layer to determine the optimal mapping need not add to encoder complexity. For coding efficiency reasons, we enforce monotonicity in each dimension of the table. Thus we need only record the starting value and step points for a reasonably small number of values. The coding process is the same as for EOB offsets.

For refinement bits, because a value of zero is statistically more probable, we assign codewords based upon the number of zeros in each group of three refinement bits, shown in Table 2.

Symbol	Codeword
0000	00
0001	010
0010	011
0011	11000
0100	100
0101	11001
0110	11010
0111	111100
1000	101
1001	11011
1010	11100
1011	111101
1100	11101
1101	111110
1110	1111110
1111	1111111

Table 2

Sign bits are either coded as flags (in the case of the second and subsequent FGS layers), or using the above grouping approach (in the case of the first FGS layer).

Results

As entropy coding affects rate and not PSNR, tables are presented showing the overhead associated with CAVLC compared to CABAC.

Sequence	Spatial (QCIF-CIF)	FGS
Bus	10.7%	10.5%
Football	5.9%	6.8%
Foreman	16.1%	10.2%
Mobile	12.4%	11.1%

Results were not generated for the 4CIF cases. CAVLC is known to perform worse on such larger resolutions, and we see that trend in the spatial results above. However, minimal effort was spent in tuning CAVLC for spatial scalability, leaving room for future improvement. While there is also scope for improving the FGS results, performance gains with CAVLC usually require values to be grouped, and this is likely to have a detrimental effect on the performance when the layer is truncated.

Conclusion

CAVLC is perceived as a valuable feature that is currently missing from the JSVM. The implementation described herein allows the base layer to be compatible with the JM 9.6 entropy decoder, and furthermore demonstrates that performance will not degrade unacceptably in enhancement layers. Nokia seeks approval from the JVT to commit the attached enhancements to the CVS server, under the direction of the software's “new features” coordinator.

(Append for Proposal Documents)

JVT Patent Disclosure Form

International Telecommunication Union
Telecommunication Standardization Sector

International Organization for Standardization

International Electrotechnical Commission

Joint Video Coding Experts Group - *Patent Disclosure Form*

(Typically one per contribution and one per Standard | Recommendation)

Please send to:
JVT Rapporteur Gary Sullivan, Microsoft Corp., One Microsoft Way, Bldg. 9, Redmond WA 98052-6399, USA
Email (preferred): Gary.Sullivan@itu.int Fax: +1 425 706 7329 (+1 425 70MSFAX)

This form provides the ITU-T | ISO/IEC Joint Video Coding Experts Group (JVT) with information about the patent status of techniques used in or proposed for incorporation in a Recommendation | Standard. JVT requires that all technical contributions be accompanied with this form. *Anyone* with knowledge of any patent affecting the use of JVT work, of their own or of any other entity ("third parties"), is strongly encouraged to submit this form as well.

This information will be maintained in a "living list" by JVT during the progress of their work, on a best effort basis. If a given technical proposal is not incorporated in a Recommendation | Standard, the relevant patent information will be removed from the "living list". The intent is that the JVT experts should know in advance of any patent issues with particular proposals or techniques, so that these may be addressed well before final approval.

This is not a binding legal document; it is provided to JVT for information only, on a best effort, good faith basis. Please submit corrected or updated forms if your knowledge or situation changes.

This form is *not* a substitute for the *ITU ISO IEC Patent Statement and Licensing Declaration*, which should be submitted by Patent Holders to the ITU TSB Director and ISO Secretary General before final approval.

Submitting Organization or Person:

Organization name	**Nokia, Inc.**
Mailing address	**6000 Connection Dr., MS 3-4-1400** **Irving, TX 75039**
Country	**USA**
Contact person	**Justin Ridge**
Telephone	+1 972 374 0628
Fax	
Email	Justin.Ridge @ nokia.com
Place and date of submission	21 July 2005, Poznan, Poland

Relevant Recommendation | Standard and, if applicable, Contribution:

Name (ex: "JVT")	
Title	
Contribution number	

(Form continues on next page)

Disclosure information – Submitting Organization/Person (choose one box)

☐ 2.0 The submitter is not aware of having any granted, pending, or planned patents associated with the technical content of the Recommendation | Standard or Contribution.

or,

The submitter (Patent Holder) has granted, pending, or planned patents associated with the technical content of the Recommendation | Standard or Contribution. In which case,

☐ 2.1 The Patent Holder is prepared to grant – on the basis of reciprocity for the above Recommendation | Standard – a free license to an unrestricted number of applicants on a worldwide, non-discriminatory basis to manufacture, use and/or sell implementations of the above Recommendation | Standard.

☐ 2.2 The Patent Holder is prepared to grant – on the basis of reciprocity for the above Recommendation | Standard – a license to an unrestricted number of applicants on a worldwide, non-discriminatory basis and on reasonable terms and conditions to manufacture, use and/ or sell implementations of the above Recommendation | Standard.

Such negotiations are left to the parties concerned and are performed outside the ITU | ISO/IEC.

X 2.2.1 The same as box 2.2 above, but in addition the Patent Holder is prepared to grant a “royalty-free” license to anyone on condition that all other patent holders do the same.

☐ 2.3 The Patent Holder is unwilling to grant licenses according to the provisions of either 2.1, 2.2, or 2.2.1 above. In this case, the following information must be provided as part of this declaration:

- patent registration/application number;
- an indication of which portions of the Recommendation | Standard are affected.
- a description of the patent claims covering the Recommendation | Standard;

In the case of any box ***other than 2.0*** *above, please provide the following:*

Patent number(s)/status ____________________________________

Inventor(s)/Assignee(s) ____________________________________

Relevance to JVT ____________________________________

Any other remarks: ____________________________________

(please provide attachments if more space is needed)

(form continues on next page)

Third party patent information – fill in based on your best knowledge of relevant patents granted, pending, or planned by other people or by organizations other than your own.

Disclosure information – Third Party Patents (choose one box)

☐ 3.1 The submitter is not aware of any granted, pending, or planned patents *held by third parties* associated with the technical content of the Recommendation | Standard or Contribution.

☐ 3.2 The submitter believes third parties may have granted, pending, or planned patents associated with the technical content of the Recommendation | Standard or Contribution.

For box 3.2, please provide as much information as is known (provide attachments if more space needed) - JVT will attempt to contact third parties to obtain more information:

3rd party name(s) ______________________________

Mailing address ______________________________
Country ______________________________
Contact person ______________________________
Telephone ______________________________
Fax ______________________________
Email ______________________________
Patent number/status ______________________________
Inventor/Assignee ______________________________
Relevance to JVT ______________________________

Any other comments or remarks: