JOINT POSITION AND AMPLITUDE CODING IN HYBRID VARIABLE LENGTH CODING FOR VIDEO COMPRESSION

Junlin Li and Ghassan AlRegib

Georgia Institute of Technology
{lijunlin, gregib}@ece.gatech.edu

Dihong Tian, Pi Sheng Chang and Wen H. Chen

Cisco Systems, Inc.
{dtian, pschang, wenc}@cisco.com

ABSTRACT

Hybrid variable length coding (HVLC) was recently proposed as a novel entropy coding scheme for block-based image and video compression, which divides each transform block into low frequency (LF) region and high frequency (HF) region and codes them differently. To efficiently code LF region, a two-dimensional position and one-dimensional amplitude coding scheme (2DP1DA) was also proposed, which jointly codes the 2D position information, i.e., run of consecutive zero-valued coefficients and run of consecutive nonzero coefficients. To further explore the potential of HVLC concept, we propose a new scheme for coding LF region, which codes the 2D position and amplitude information of each nonzero cluster jointly with manageable complexity. The experimental results show that compared with CAVLC in H.264, about 3.5% bit rate reduction is achieved by the proposed method for a wide range of quantization parameters (QP).

Index Terms— Video coding, entropy coding, VLC, H.264.

1. INTRODUCTION

Existing video coding standards, such as MPEG-2/4, H.263, and H.264/AVC [1], commonly adopt the so-called block-based hybrid video coding approach, where motion-compensation prediction is used to exploit the temporal redundancy, spatial prediction and/or transform coding of the prediction residual are used to exploit the spatial redundancy, and entropy coding is used to exploit the statistic redundancy of the quantized transform coefficients. Variable length coding (VLC) is widely deployed for entropy coding due to its efficiency and simplicity, where the entropy encoder assigns one variable length codeword to each of the symbols, and VLC tables are designed such that symbols appearing more often are encoded by shorter codewords, thus resulting in a short average code length.

In the context of VLC, different representations of quantized transform coefficient array are used to construct the coding symbols and thus to generate the VLC table. A conventional representation is (run, level) pair, referred to as RL-VLC, where“run” indicates the number of zeros preceding a nonzero coefficient and “level” indicates the magnitude of the nonzero coefficient. RL-VLC, adopted by H.263, is efficient to code scattered nonzero coefficients, however, it is inefficient in coding clustered nonzero coefficients, due to the fact that n separate codes are required to represent n consecutive nonzero coefficients, each of which has a run equal to zero. In the latest video coding standard, i.e., H.264/AVC, a more sophisticated context-adaptive VLC , referred to as CAVLC, is used. Similar with RL-VLC, CAVLC is inefficient in coding clustered nonzero coefficients since it also needs to code all the zero-runs separately with the exception that zero-runs at the start of the array need not be encoded. This is because they can be inferred from the already coded parameters in the block. Compared with RL-VLC, CAVLC is more efficient due to the introduction of multiple VLC tables with context adaptive table switch. Multiple tables based RL-VLC was also studied in [2, 3], where context-adaptive table switch is used in [2], while a position-dependent table switch is used in [3]. Another run-length based adaptive coding was also proposed in [4] to encode the “run” and “level” separately using adaptive binary arithmetic coding.

Hybrid variable length coding (HVLC) [5] was recently proposed for H.263 video coding, which takes advantage of the clustered nature of the quantized nonzero coefficients in the low-frequency (LF) region and the scattered nature of the quantized nonzero coefficients in the high-frequency (HF) region by employing two types of VLC schemes. Since conventional RL-VLC is efficient to code scattered nonzero coefficients, it is adopted by HVLC for coding HF region. However, new efficient schemes for coding LF region are expected to utilize the clustered nature of nonzero coefficients in the LF region. For coding LF region, two-dimensional position and one-dimensional amplitude coding (2DP1DA) scheme [5] was proposed, which codes the run of consecutive zero coefficients and run of consecutive nonzero coefficients as a pair by a two-dimensional VLC table, and codes the amplitude of each nonzero coefficient independently by a one-dimensional VLC table. As shown in [5], HVLC with 2DP1DA for coding LF region is superior to RL-VLC in H.263. In this paper, we show that HVLC with 2DP1DA for coding LF region is also superior to CAVLC in H.264. Moreover, we propose a new scheme for coding LF region, which utilizes not only the clustered nature of the nonzero coefficients in LF region but also the fact that the trailing nonzero coefficients in each nonzero cluster tend to be small. It jointly codes the 2D position and amplitude information of each nonzero cluster with manageable complexity. The experimental results show that it not only further improves the coding efficiency, but also provides a more uniform gain over a wide range of QP.

The subsequent sections are organized as follows. In Section 2 we provide an overview of the hybrid variable length coding concept. In Section 3, the proposed new scheme for coding LF region is described in detail. Then, we present experimental results in Section 4 and conclude the paper in Section 5.

2. HYBRID VARIABLE LENGTH CODING

In HVLC, a breakpoint, which is a coefficient index along the coefficient scan path, is first defined as shown in Fig. 1. The coefficients below and above the breakpoint are considered as low-frequency (LF) and high-frequency (HF) coefficients, respectively, where the nonzero coefficients are statistically more clustered in LF region while more scattered in HF region. In HVLC, the conventional RL-VLC or an equivalent scheme is used for coding the HF coefficients, while a new coding scheme is used for coding the LF coefficients to exploit the clustered nature of nonzero coefficients in LF region, such as 2DP1DA in [5] or joint position and amplitude coding (JPAC) proposed in this paper, which will be described in detail in Section 3. The breakpoint must be known to the decoder to properly decode the coefficients. To avoid using two codewords for one coefficient around the breakpoint, the breakpoint is extended beyond the LF region to the last coefficient coded by the LF coding scheme.

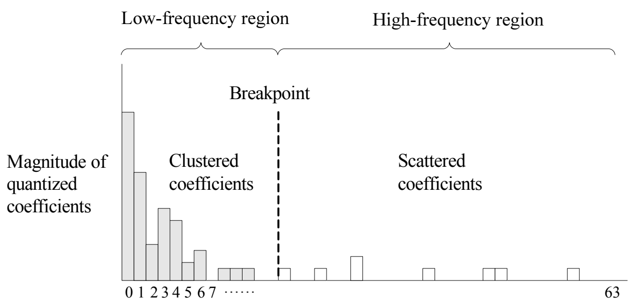


Fig. 1. Coefficient scan of an 8×8 block along a pre-defined path, e.g., zigzag.

3. JOINT POSITION AND AMPLITUDE CODING (JPAC) FOR LF REGION

There are essentially two types of information that need to be represented in coding a sequence of quantized transform coefficients: the positions of the nonzero coefficients and their corresponding amplitudes. In RL-VLC scheme, the position and amplitude of each nonzero coefficient are jointly coded as a (run, level) pair, but different nonzero coefficients are coded independently. On the contrary, in 2DP1DA scheme [5], the positions of consecutive nonzero coefficients are jointly coded as a two-dimensional position code (2DP), i.e., the run of zeros that precede the nonzero cluster and the run of nonzero coefficients within the cluster, while the amplitudes of the nonzero coefficients are coded separately from the position. To further improve the coding efficiency, it is desirable to code the positions and amplitudes of the consecutive nonzero coefficients together by a multi-dimensional code combining the 2DP information and the amplitudes of all nonzero coefficients in the cluster. From Shannon information theory, it is obvious that fully joint position and amplitude coding will definitely improve the coding efficiency, however, the complexity, such as code table size, will grow exponentially with the number of consecutive nonzero coefficients and the amplitude level of the nonzero coefficients.

As shown in Fig. 1, the nonzero coefficients are statistically more clustered in LF region, while they are scattered in HF region, which motivates the HVLC design. It is also observed that the amplitudes of the trailing coefficients in each nonzero cluster in LF region tend to be small, say 1 or -1. Thus, we design a new LF coding scheme, referred as JPAC, which takes this behavior into account and codes the 2D position information and the amplitude information jointly with a manageable code table size. To make the code table manageable, the proposed JPAC scheme customizes the joint 2D position and amplitude coding concept in twofolds:

- JPAC jointly codes only the trailing few nonzero coefficients in each nonzero cluster since only the trailing coefficients tend to be small, more specifically, only t nonzero coefficients at the end of each cluster are jointly coded, where

$$t = \min(n, M),$$

n denotes the total number of nonzero coefficients in the cluster, and M is a threshold, i.e., maximum number of nonzero coefficients to be coded jointly. Therefore, if $n \leq M$ for a given cluster, all the nonzero coefficients are jointly coded, otherwise, only the M nonzero coefficients at the end of the cluster are jointly coded, all the first $(n - M)$ coefficients are coded separately with a one-dimensional VLC table.

- JPAC jointly codes the amplitude of each trailing nonzero coefficient as a binary information, that is '1' or 'non 1', to take advantage of the most frequent appearance of the coefficient level '1'. If the magnitude is '1', no additional coding is necessary, otherwise, its amplitude minus 1 is coded using a one-dimensional VLC table.

Based on the design above, the coding symbols of JPAC are shown in Table 1, where we assume $M = 3$, and 'x' represents a nonzero coefficient that has a magnitude larger than 1.

Since a run of nonzero coefficients implies that the following coefficient is a zero-valued coefficient (otherwise it would have been counted into the nonzero cluster), it can be skipped in the coding process to save bits. Similar to H.263 and 2DP1DA [5], a binary information, '0' or '1', is also incorporated into the JPAC symbol for LF region or RL symbol for HF region to indicate whether this is the last nonzero coefficient in the block.

To illustrate the proposed HVLC coding scheme using JPAC for LF region and RL-VLC for HF region, consider an example shown in Table 2, where all the remaining coefficients in the sequences are assumed to be zeros, and a constant breakpoint $N = 14$ and a maximum threshold $M = 3$ are used. In the coding stream, C_{PA} denotes the JPAC codeword, C_A denotes the one-dimensional amplitude codeword, and C_{RL} denotes the codeword of RL-VLC. Note that the sign of each nonzero coefficient is coded separately with 1 bit and omitted in Table 2.

It should be noted that 2DP1DA in [5] is an extreme case of the proposed JPAC scheme with $M = 0$. Another extreme case of JPAC is that when M is large enough such that all the nonzero coefficients are jointly coded, for example, $M = 64$ for 8×8 block. On the other hand, the code table size will grow exponentially with M. So there exists an optimal M to achieve the best tradeoff between the coding efficiency and the code table size, which will be determined experimentally in Section 4.

4. EXPERIMENTAL RESULTS

In this section, we report preliminary test results of HVLC using JPAC with $M = 3$ for coding LF region and RL-VLC for coding HF region. We incorporate HVLC into the H.264 codec by replacing the CAVLC entropy coding scheme in H.264 with HVLC scheme. Since the proposed HVLC algorithm is targeted at coding high resolution video sequences with 8×8 transform block size, H.264/AVC FRExt with fixed 8×8 transform is used for both HVLC and CAVLC in the simulation.

The test database includes ten 4CIF standard test sequences. Each sequence has 210 frames with a frame rate of 25 frames per second (fps). The GOP length is 15 frames, where one is INTRA coded and the others are INTER coded. The HVLC code tables are

Table 1. Symbols of JPAC

zero run	nonzero run with amplitudes of the trailing 3 nonzero coefficients													
	1		2				3			...	n			...
0	(0,1,1)	(0,1,x)	(0,2,11)	(0,2,1x)	(0,2,x1)	(0,2,xx)	(0,3,111)	...	(0,3,xxx)	...	(0,n,111)	...	(0,n,xxx)	...
1	(1,1,1)	(1,1,x)	(1,2,11)	(1,2,1x)	(1,2,x1)	(1,2,xx)	(1,3,111)	...	(1,3,xxx)	...	(1,n,111)	...	(1,n,xxx)	...
2	(2,1,1)	(2,1,x)	(2,2,11)	(2,2,1x)	(2,2,x1)	(2,2,xx)	(2,3,111)	...	(2,3,xxx)	...	(2,n,111)	...	(2,n,xxx)	...
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
z	(z,1,1)	(z,1,x)	(z,2,11)	(z,2,1x)	(z,2,x1)	(z,2,xx)	(z,3,111)	...	(z,3,xxx)	...	(z,n,111)	...	(z,n,xxx)	...
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...

Table 2. An Example of HVLC with JPAC for LF region and RL-VLC for HF region

Index:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	$\cdots$
Coefficient:	9	-5	3	-2	1	0	0	2	1	1	0	0	0	-1	1	0	0	0	1	0	$\cdots$
Coding Stream:	$C_{PA}(0,5,\text{xx1},0)$ $C_A(9)$ $C_A(5)$ $C_A(2)$ $C_A(1)$ $C_{PA}(1,3,\text{x11},0)$ $C_A(1)$ $C_{PA}(2,2,11,0)$ $C_{RL}(2,1,1)$																				

constructed based on the measured statistics of symbols and are generated separately for INTRA and INTER modes. A constant breakpoint (N) is used for a given quantization parameter (QP). In particular, we use three pairs of (QP, N) parameters, i.e., (QP=5, N=27), (QP=25, N=20), and (QP=37, N=14), as test examples for small, medium, and large QPs respectively, and their corresponding breakpoints are determined by gathering the statistics from the training sequences to appropriately separate the clustered region and scattered region. For comparison purpose, HVLC with 2DP1DA for coding LF region and CAVLC in H.264 are also tested. Note that the complexity of the proposed algorithm and CAVLC in H.264 is much less than the complexity of CABAC in H.264 [6], therefore, CABAC is not included in the comparison. The bit-rate results for coding the test sequences are shown in Table 3, 4 and 5, respectively. All the bit-rate results are in kbits/sec. A positive percentage in the tables indicates the bit-rate reduction achieved by HVLC, while a negative percentage indicates that the bit rate is increased by HVLC.

From the results, it can be seen that compared with CAVLC, HVLC with 2DP1DA can reduce the bit rate averagely by 2.9% for QP=5, 2.0% for QP=25 and only 0.5% for QP=37, i.e., a larger gain for small QP, while a marginal gain for large QP. It can be easily understood since there are more nonzero coefficients and clusters in LF region when QP is small, and vice versa. By jointly coding the 2D position and amplitude, JPAC can further reduce the bit rate considerably and the gain is quite uniform for a wide range of QP, i.e., 3.2% for QP=5, 3.7% for QP=25 and 3.5% for QP=37. It can be understood as follows: for small QP, a big gain is achieved by solo 2D position coding while a small additional gain is achieved by joint amplitude with position coding since there are many large nonzero clusters and most of the nonzero coefficients tend to be large; on the contrary, for large QP, a small gain is achieved by solo 2D position coding while a big additional gain is achieved by joint amplitude with position coding since the nonzero clusters tend to be small and most of the nonzero coefficients tend to be small too.

Furthermore, to determine the optimal threshold M, i.e., the optimal number of jointly coded nonzero coefficients, several candidates ($M = 1, \cdots, 5$) are tested. The two extreme cases of $M = 0$ and $M = 64$ are also tested for comparison purpose. For the case of $M = 64$, we constraint the maximum nonzero run to 10 and divide the clusters larger than 10 into small sub-clusters to make the code table manageable. The bit rate reduction results for different M are shown in Fig. 2($a \sim c$) for QP=5, 25, and 37, respectively. From Fig. 2, we can see that for both INTRA and INTER modes and different QP, $M = 3$ achieves a reasonable tradeoff between the coding efficiency and the code table size. It is also noted that in JPAC scheme, joint coding all the nonzero coefficients, i.e., $M = 64$, does not necessarily provide the best coding performance, especially for small QP, such as QP=5, because there are many large clusters and most of the nonzero coefficients are far larger than 1 in LF region.

5. CONCLUSIONS AND FUTURE WORK

By jointly coding the 2D position and amplitude information of each nonzero cluster, HVLC with JPAC for coding LF region achieves about 3.5% bit rate reduction for a wide range of quantization parameters (QP). JPAC takes advantage of the clustered nature of the nonzero coefficients in the LF region and the fact that the trailing nonzero coefficients in each cluster tend to be small.

In this paper, a fixed breakpoint for a given QP is used, which is not necessarily optimal. Also empirical studies show that it is difficult to find a common "optimal" breakpoint for a given coding parameter since the optimal breakpoint varies with different sequences, even different blocks in the same sequence. One way to solve this problem is to include the optimal breakpoint for each individual block in the bitstream, which introduces a considerable overhead. Another way is to design a context-adaptive mechanism to determine the "optimal" breakpoint for each block automatically without any extra cost, which is our future research direction.

6. REFERENCES

[1] ISO/IEC 14496-10, *Draft of Version 4 of H.264/AVC*, 2005.

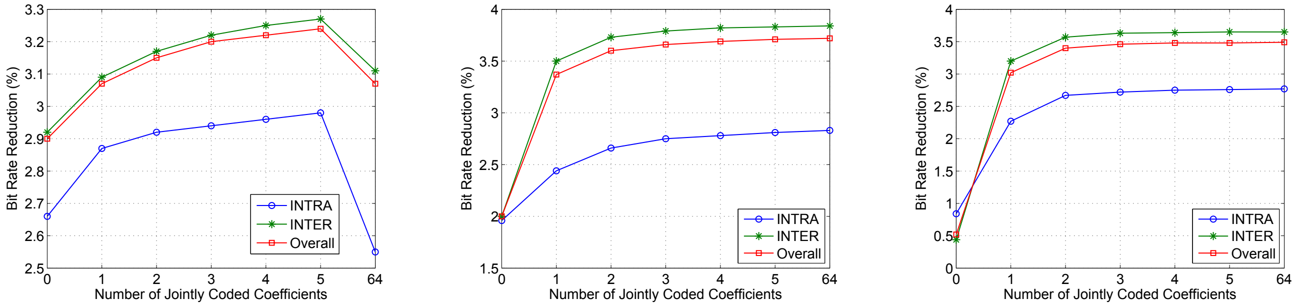
[2] Q. Wang, D.-B. Zhao, and W. Gao, "Context-based 2D-VLC entropy coder in AVS video coding standard," *J. Comput. Sci. Technol.*, vol. 21, no. 3, pp. 315–322, 2006.

[3] G. Lakhani, "Optimal huffman coding for dct blocks," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 14, no. 4, pp. 522–527, Apr. 2004.

[4] C. Tu, J. Liang, and T. Tran, "Adaptive runlength coding," *IEEE Signal Processing Letters*, vol. 10, no. 3, pp. 61–64, Mar. 2003.

[5] D. Tian, W. H. Chen, P. S. Chang, G. AlRegib, and R. Mersereau, "Hybrid variable length coding for image and video compression," in *Proc. of the Intl. Conf. on Acoustics, Speech, and Signal Processing*, Honolulu, HI, Apr. 15-20 2007, vol. 1, pp. 1133–1136.

[6] D. Marpe, H. Schwarz, and T. Wiegand, "Context-Based Adaptive Binary Arithmetic Coding in the H.264/AVC Video Compression Standard," *IEEE Trans. Circuits and Systems for Video Tehnology*, vol. 13, no. 7, pp. 620–636, Jul. 2003.



(a) (b) (c)

Fig. 2. Bit Rate Reduction for different number of jointly coded coefficients: (a) QP=5, (b) QP=25, and (c) QP=37.

Table 3. Bit-rate results for QP = 5 and $N = 27$ (kb/s)

Sequence	INTRA			INTER			Overall		
	H.264[a]	2DP1DA	JPAC	H.264	2DP1DA	JPAC	H.264	2DP1DA	JPAC
Crew	3688.02	0.28%	0.53%	47765.26	0.35%	0.72%	51453.28	0.34%	0.71%
Harbour	4319.22	2.92%	3.09%	51629.02	1.15%	1.22%	55948.23	1.29%	1.36%
Soccer	3667.38	2.50%	2.87%	43490.45	1.31%	1.86%	47157.83	1.40%	1.94%
Barcelona	7095.85	-0.41%	-0.25%	75807.66	3.78%	3.93%	82903.51	3.42%	3.57%
Fries	3820.64	0.88%	1.10%	47980.86	0.40%	0.97%	51801.5	0.43%	0.98%
Mobile	6499.55	0.39%	0.79%	64043.72	3.03%	3.42%	70543.27	2.78%	3.18%
Music	5318.01	1.11%	1.34%	62068.96	3.14%	3.24%	67386.97	2.98%	3.09%
Race	5145.37	3.59%	3.94%	63949.22	3.49%	3.80%	69094.59	3.49%	3.81%
Rower	4726.3	9.47%	9.91%	56193.65	5.27%	5.73%	60919.96	5.59%	6.06%
Rugby	5114.3	7.12%	7.40%	64950.39	5.25%	5.42%	70064.69	5.39%	5.57%
Total	49394.64	2.66%	2.94%	577879.19	2.92%	3.22%	627273.83	2.90%	3.20%

Table 4. Bit-rate results for QP = 25 and $N = 20$ (kb/s)

Sequence	INTRA			INTER			Overall		
	H.264	2DP1DA	JPAC	H.264	2DP1DA	JPAC	H.264	2DP1DA	JPAC
Crew	573.12	-0.23%	1.90%	4444	-2.26%	1.17%	5017.12	-2.03%	1.25%
Harbour	1047.71	5.16%	6.13%	7407.56	-1.32%	2.54%	8455.27	-0.52%	2.98%
Soccer	782.68	1.76%	2.90%	4745.95	-1.04%	2.32%	5528.63	-0.64%	2.40%
Barcelona	2336.33	-1.47%	-1.03%	13491.89	1.40%	3.42%	15828.22	0.98%	2.76%
Fries	539.73	0.52%	2.58%	4247.23	-1.60%	1.31%	4786.96	-1.36%	1.45%
Mobile	2081.04	-1.91%	-1.33%	10822.53	1.43%	3.37%	12903.57	0.89%	2.62%
Music	1253.53	-2.55%	-1.66%	8293.97	-0.10%	1.49%	9547.49	-0.42%	1.08%
Race	1349.74	3.98%	4.40%	13559.64	4.53%	4.93%	14909.37	4.48%	4.89%
Rower	1138.95	10.66%	11.31%	9767.34	5.68%	6.55%	10906.29	6.20%	7.05%
Rugby	1293.31	8.08%	8.81%	13304.42	4.63%	5.62%	14597.73	4.94%	5.90%
Total	12396.14	1.96%	2.75%	90084.53	2.00%	3.79%	102480.65	2.00%	3.66%

Table 5. Bit-rate results for QP = 37 and $N = 14$ (kb/s)

Sequence	INTRA			INTER			Overall		
	H.264	2DP1DA	JPAC	H.264	2DP1DA	JPAC	H.264	2DP1DA	JPAC
Crew	145.63	0.06%	2.89%	754.14	-0.11%	1.84%	899.77	-0.08%	2.01%
Harbour	332.74	2.80%	6.21%	1073.7	-3.23%	2.14%	1406.44	-1.80%	3.11%
Soccer	188.96	-0.74%	2.44%	790.36	-0.63%	2.31%	979.32	-0.65%	2.33%
Barcelona	734.38	-0.65%	0.77%	1645.03	-1.01%	2.08%	2379.42	-0.90%	1.68%
Fries	151.49	1.87%	4.47%	906.1	0.14%	2.49%	1057.59	0.39%	2.77%
Mobile	717.25	-1.28%	-0.50%	1736.41	0.07%	3.59%	2453.67	-0.32%	2.39%
Music	400.12	-1.90%	-1.02%	1630.86	-0.92%	2.22%	2030.99	-1.11%	1.58%
Race	410.72	1.73%	3.63%	2867.58	0.22%	3.79%	3278.3	0.41%	3.77%
Rower	347.09	4.43%	7.54%	2191.23	1.08%	4.46%	2538.31	1.53%	4.88%
Rugby	390.92	5.19%	7.36%	3123.91	3.64%	6.09%	3514.83	3.81%	6.23%
Total	3819.3	0.84%	2.72%	16719.32	0.44%	3.63%	20538.64	0.52%	3.46%

[a] In Tables 3, 4 and 5, the results of H.264 are generated using H.264/AVC FRExt with CAVLC and fixed 8×8 transform.