ADDITIONAL SIGN BIT HIDING OF TRANSFORM COEFFICIENTS IN HEVC

Xingyu ZHANG, Oscar C. AU*, Chao PANG*, Wei DAI*, Yuanfang GUO*, Lu FANG†*

*The Hong Kong University of Science and Technology
{eexyzhang, eeau, pcece, weida, eeandyguo}@ust.hk
†University of Science and Technology of China, fanglu@ustc.edu.cn

ABSTRACT

Transform coefficients take up a significant portion of the transmitted bitstream generated by video codec. For every non-zero coefficient, the sign information is sent separately from the amplitude. According to the statistical patterns, the signs of the transform coefficients are rarely predictable, thus bypass mode is usually used in the following CABAC engine to encode the sign information. In H.265/HEVC, a novel multiple sign bits hiding (MSBH) technique is adopted to hide one sign for each selected CG. In this paper, we propose to further perform one additional sign bit hiding (ASBH) for the selected TBs while the MSBH can coexist. To verify the effectiveness of our proposed ASBH, we implemented it on the state-of-the-art HM9.0 test model of H.265/HEVC. Experimental results show that the proposed ASBH always outperforms the anchor for all tested sequences with very trivial complexity added at the decoder side.

Index Terms— H.265/HEVC, Video coding, transform coefficients coding, sign bit hiding, data hiding.

1. INTRODUCTION

H.265/HEVC is the state-of-the-art video coding standard which is finalized in January 2013. It's a joint work by the ITU-T and ISO/IEC people in a collaborative team named JCT-VC. H.265/HEVC follows the hybrid video coding structure as its predecessors. Thanks to the various tools that H.265/HEVC has adopted, its compression performance significantly outperforms other existing advanced standards by about 50% bit rate reduction while keeping equal perceptual video quality[1].

A distinguished feature of H.265/HEVC is its support of large blocks and block subpartition for coding, prediction, and transform processes. Especially for transform, the transform block (TB) can be of size 4x4, 8x8, 16x16 and 32x32. In order to encode the coefficients of large TB, the concept of coefficient group (CG) is used[2]. A CG is an array of 16 quantized coefficients counted in a certain scanning order (horizontal, vertical or diagonal). Therefore a TB of size NxN will have $\frac{N^2}{16}$ CGs. Fig. 1 shows an example of the CG parition in a 16x16 TB, each 4x4 square with the black boundary lines is a CG. The CG filled by blue color means it includes at least one non-zero coefficient. We call the blue CG *significant CG* in this paper. The CG filled by white color means all coefficients in that CG are zero. We call the white CG *empty CG*.

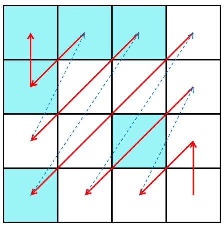

Fig. 1. An example of the CG parition in a 16x16 TB.

Data hiding is a very critical technique in the field of fingerprinting, error concealment, and copyright protection etc. In recent years, researchers have tried to use the data hiding technique in video coding. In [3], it proposes to hide the index of the motion vector predictor in the chroma and luma transform coefficients through a RD optimization. In [4], the most probable mode index for intra prediction is proposed to be inferred from the parity of the sum of the quantized coefficients. In [5], it proposes to embed the selection of transform modes into the parity of the sum of the quantized coefficients. The common aspects among [3, 4, 5] are: (1) they all hide a binary syntax into a binary function; (2) they all use the quantized coefficients as the target source for data hiding. A similar but slightly different idea is also proposed in [6] to embed the binary syntax of motion vector predictor selection into a parity function of the motion vector differences rather than the quantized coefficients. During the development of H.265/HEVC, a successful design of data hiding technique is adopted as proposed in [7]. It embeds one sign bit into the parity of the sum of the quantization coefficients at CG level conditionally. Therefore, multiple sign bits might be hidden when the large TB has multiple significant CGs.

In this paper, we propose to further hide one additional sign bit of the transform coefficients at TB level conditionally. Different from the above mentioned works, our proposed

---

This work has been supported in part by the Research Grants Council (GRF Project no. 610112) and the HKUST (HKUST Project no. FS-GRF12EG01) of the Hong Kong Special Administrative Region.

ASBH divides the TB coefficients into two arrays: an even array and an odd array. We design to embed an additional sign bit into the parity of the sum of the TB even array, and let the odd array play an role of coordination between the proposed ASBH and the existing MSBH.

The rest of this paper is organized as follows. In section 2, the adopted MSBH in H.265/HEVC is revisited first. Section 3 introduces the detail of our proposed ASBH technique with an discussion on the encoder implementation. Finally, experimental results are shown in Section 4, with a conclusion drawn in Section 5.

2. MSBH IN H.265/HEVC

In a TB larger than 4x4, the quantized transform coefficients are encoded in a two-layer manner. In the first layer, a significant CG map comprised of a binary string is encoded to indicate the significant CGs and the empty CGs. In the second layer, only the significant CGs will encode the individual quantized coefficients. As an example shown in Fig.1, in the second layer, the blue CGs will encode the significant coefficient map (zero/non-zero coefficients), the magnitude of the non-zero coefficient, and the sign of the non-zero coefficient. However all coefficients in the white CGs will be skipped in the second layer and inferred to be zero directly. With respect to the sign information, a ”0” bit is sent for a positive quantized coefficient, and a ”1” bit is sent for a negative quantized coefficient. In the multiple sign bits hiding (MSBH) scheme, the quantized coefficients in a significant CG are used as the target source to hide the first sign bit in the corresponding CG. When the TB is of large size and consists of multiple significant CGs, multiple first sign bits might be hidden.

In the decoder, the condition of inferring the first sign bit depends on the positions of the first and last non-zero coefficients. If there are more than T_1 (zero/non-zero) coefficients between the first and the last non-zero coefficients, the MSBH condition is satisfied. The first sign bit in this CG is inferred to be positive when the parity of the sum of the CG quantized coefficients is even, and negative otherwise. When the MSBH condition disconfirms, the first sign bit is just decoded in the bypass mode as other sign bits.

To match the decoder, the encoder needs to check the number of coefficients between the first and the last non-zero coefficients in a significant CG as well. If the MSBH condition is not satisfied, the first sign bit of the CG is just encoded and sent. Otherwise, the first sign bit in this CG is saved from encoding. When the saved sign bit and the parity does not match, the encoder needs to adjust one quantized coefficients in the corresponding CG by +1 or -1.

3. PROPOSED ASBH

In our proposed additional sign bit hiding (ASBH) technique, we want to further exploit the potential of the quantized coefficients as the target source for data hiding. We tap the potential by embedding additional information into the quantized coefficients at a different level from the CG level. To be specific, we divide all the quantized coefficients in a TB into two arrays. One comprises the TB quantized coefficients at even positions in the scanning order, i.e. TB even array, as the the blue coefficients in Fig. 2. The other one comprises the TB quantized coefficients at odd positions in the scanning order, i.e. TB odd array, as the the yellow coefficients in Fig. 2. While the TB even array is used as the target source, the TB odd array is left as tuning flexibility to avoid conflicts between the TB level data hiding and the CG level data hiding.

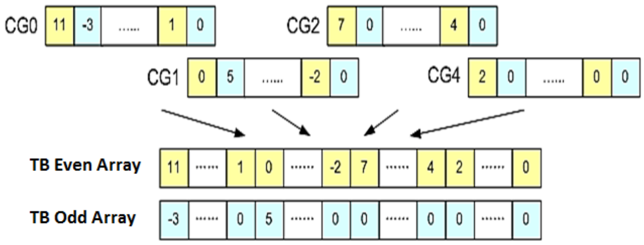

Fig. 2. An example of the TB even array and TB odd array.

As for the additional syntax to be hidden, it favors the following three properties. First, the possible values of this syntax are equiprobable. Second, the syntax is uncorrelated with the target source. Third, the syntax takes a considerable proportion of the transmitted bits. A syntax with the first and the second properties is very bits-consuming in the later CABAC entropy coding engine. And saving the bits of encoding a syntax with the third property will lead to noticeable bit rate reduction. Considering all three properties, we select one sign of the quantized coefficient as the additional syntax to hide, similar to the existing sign bit hiding technique. To be more specific, if MSBH is applied to the first significant CG, the *additional hiding sign* is the sign of the second non-zero coefficient in the TB. Otherwise, if MSBH is not applied to the first significant CG, the additional hiding sign is the sign of the first non-zero coefficient in the TB.

3.1. ASBH at the decoder

For each TB, the decoder first decodes the quantized coefficients CG by CG in a reverse path of the scanning order as H.265/HEVC defines, i.e. from the last CG to the first CG as described by the red line in Fig.1. When the first significant CG comes to be decoded, the number of the significant CGs in the TB has been available. If the number is equal or larger than a threshold T_2, the additional sign bit hiding (ASBH) condition is satisfied. The additional hiding sign is inferred from the parity of the sum of the TB even array. Even parity indicates a positive sign, and odd parity indicates a negative sign. Otherwise, the first sign bit and the second sign bit are just inferred or decoded as H.265/HEVC defines. The reason for counting the number of significant CGs is to allow the encoder matching the decoder at a moderate RD loss, since a number of coefficients can be the candidates for adjustment when necessary. Obviously, our proposed ASBH causes very little latency at the decoder side.

3.2. ASBH at the encoder

To match the decoder, the encoder needs to check the number of significant CGs in the TB. When the ASBH condition is not met, no additional sign bit will be saved. When the ASBH condition is satisfied, the additional sign bit is saved and needs to match the parity of the sum of the to-be-encoded TB even array. At the same time, since our proposed ASBH can be applied to H.265/HEVC, the MSBH condition for each significant CG should be checked as well. We need to consider ASBH and MSBH jointly.

After the rate distortion optimized quantization (RDOQ) [8] process, the significant CGs can be classified into three sets. The first set S_1 contains all MSBH-active CGs, which means MSBH condition is satisfied in the ith CG, denoted as CG_i with $i \in S_1$, but CG_i needs adjustment for parity match. The second set S_2 contains MSBH-silent CGs, which means the MSBH condition is satisfied in CG_i with $i \in S_2$, no adjustment is required for CG_i. The third set S_3 includes MSBH-invalid CGs, which means the MSBH condition is not met in any CGs in S_3.

Let's check how the MSBH-active CG affects the ASBH first. Consider one original coefficient c_r, which becomes u_r after RDOQ process. Adjusting u_r by δ_r, the corresponding RD cost change $J(u_r, u_r+\delta_r)$ can be formulated as

$$\begin{aligned} J(u_r, u_r+\delta_r) = D(c_r, Q^{-1}(u_r+\delta_r)) - D(c_r, Q^{-1}(u_r)) \\ + \lambda[R(u_r+\delta_r) - R(u_r)]. \qquad (1) \end{aligned}$$

Here $Q^{-1}(x)$ means the de-quantization process. The function $D(x, y)$ means the mean square difference between x and y. $R(x_r)$ means the bits used to encode the coefficient x_r at the position r. The adjustment quantity that we will consider to apply on u_r is δ^*_r

$$\delta^*_r = \arg \min_{\delta_r \in \{1,-1\}} J(u_r, u_r+\delta_r). \qquad (2)$$

For any CG_i, $i \in S_1$, we check the positions in a subset P_i, which contains all the feasible positions in CG_i that an adjustment won't change the sign of the first non-zero coefficient after updating. Then we can find the position of the coefficient that gives the minimal adjustment cost. i.e.

$$r^*_i = \arg \min_{r \in P_i} J(u_r, u_r+\delta^*_r), i \in S_1. \qquad (3)$$

During this checking process, we can find a best even position e^*_i that gives the minimal adjustment cost among all coefficients at even positions. Similarly, the best odd position d^*_i is attainable.

If the first significant CG (denoted as CG_h) is MSBH-active, CG_h is updated first according to r^*_h. After that, the additional sign bit $sign_a$ we intend to hide in our ASBH is decided by the second non-zero coefficient in CG_h if $h \in \{S_1 \cup S_2\}$, or the first non-zero coefficient if $h \in S_3$. Let

$$W = \sum_{r=0}^{N^2/2-1} |u_r| + mod(\sum_{i \in S_1} |\delta^*_{r^*_i}|(r^*_i+1), 2). \qquad (4)$$

The value of $mod(W, 2)$ indicates the parity of the sum of the TB even array if all CGs in set S_1 are updated according to their r^*_i. Therefore, if $sign_a = mod(W, 2)$, both MSBH and ASBH are satisfied after the adjustments according to r^*_i, $i \in S_1$. No additional effort for ASBH is needed.

If $sign_a \neq mod(W, 2)$, every significant CG will try to provide some solutions. An exception is for CG_h if $h \in S_1$. Finally, the solution with the minimal additional RD cost will be selected as the best solution. Let's denote the first sign bit in the ith CG_i as $sign1_i$ before adjustment and as $sign1'_i$ after adjustment; similarly, we have $sign2_i$ and $sign2'_i$ for the second sign bit. The solutions of the CG in different sets are illustrated in several scenarios in the following.

Consider a CG_i in set S_1, $i \neq h$. there are two scenarios.

- Scenario 1: CG_i adjusts its d^*_i or e^*_i instead of its r^*_i. After the adjustment, CG_i turns to be MSBH-silent. $sign1_i = sign1'_i$ is required.
- Scenario 2: CG_i adjusts one coefficient at an even position. After the adjustment, CG_i turns to be MSBH-invalid. $sign1_i = sign1'_i$ is not required.

For CG_i in set S_1, note that the scenario 2 does not always exist. In scenario 2, the only two possible candidate even positions should be at the position of the first non zero coefficient and the position of the last non zero coefficient.

Consider a CG_i in set $S2$. There are three scenarios.

- Scenario 1: CG_i adjusts two coefficients at both even and odd positions at the same time, so that CG_i remains to be MSBH-silent. $sign1_i = sign1'_i$ is required. If $i = h$, $sign2_i = sign2'_i$ is required.
- Scenario 2: CG_i adjusts one coefficient at an even position. After the adjustment, CG_i turns to be MSBH-invalid. $sign1_i = sign1'_i$ is not required. If $i = h$, $sign1'_i = sign2_i$ is required.
- Scenario 3: If the first coefficient in CG_i is zero, CG_i adjusts one zero coefficient at an even position before the first non-zero coefficient, so that the MSBH-silent situation remains. But $sign1'_i \neq sign1_i$ is required. If $i = h$, $sign1 = sign2_i = sign2'_i$ is required.

For CG_i in set $S2$, if the combination of e^*_i and d^*_i does not fit in scenario 1, other combinations are tried in the order of small to large additional RD cost until scenario 1 is satisfied.

Consider a CG_i in set $S3$. there are two scenarios.

- Scenario 1: CG_i adjusts one coefficient at an even position and remains to be MSBH-invalid. If $i = h$, $sign1'_i = sign1_i$ is required.

- Scenario 2: CG_i adjusts one coefficient at an even position. After the adjustment, CG_i turns to be MSBH-silent. $sign1'_i = [(\sum_{r=16i}^{16i+15} u_r) + 1]\%2$ is required. If $i = h$, $sign2'_i = sign1_i$ is required.

For CG_i in set $S3$, note that either scenario 1 or scenario 2 does not always exist.

We define $E1_i$ as the minimal additional cost that CG_i pays when CG_i uses its scenario 1 solution to cater the coordination between ASBH and MSBH. Similarly, $E2_i$ and $E3_i$ are defined for the minimal additional cost of scenario 2 and 3 of CG_i. The additional cost for an adjustment is calculated according to Eqn.1, except the feasible position r should be under the sign constraints of the corresponding scenarios. For CGs in subset S_1 and S_3, $E3_i = +\infty$ since there is no scenario 3 for them. For any scenario with no feasible solution in a CG, the corresponding additional cost for that scenario is then set to $+\infty$. Note that, for CG_i in subset S_1, the additional cost of scenario 1 is $E1_i = |J(u_{e_i^*}, u_{e_i^*} + \delta^*_{e_i^*}) - J(u_{d_i^*}, u_{d_i^*} + \delta^*_{d_i^*})|$. This is because a MSBH-active CG requires one coefficient change by the MSBH-active nature. In the contrast, for CGs in set S_2, the additional RD cost should be the summation of the additional RD cost of both even and odd coefficients.

Finally, the to-be-adjusted CG is decided by comparing the additional costs of all considered CGs as

$$i^* = \arg\min_{i \in A}\{E1_i, E2_i, E3_i\} \tag{5}$$

An adjustment in CG_{i^*} is used upon which scenario provides the minimal addition cost for that CG. The set A includes all the considered CGs during this adjustment selection process. For CGs with $i \in S_1$ and $i \neq i^*$, the CG just adjusts its coefficient at position r_i^* by $\delta^*_{r_i^*}$. After all these modifications, both the MSBH and ASBH are satisfied. There will be no conflicts or mismatch between the encoder and the decoder.

4. EXPERIMENTAL RESULTS

To evaluate the coding efficiency, the proposed scheme is implemented on the state-of-the-art H.265/HEVC reference software HM9.0. Intra-Main test condition and anchor configurations follow [9]. The tested QPs are 22, 27, 32 and 37. In the anchor, MSBH is on and T_1 equals to 3 by default. The T_2 in our proposed ASBH is set to 3 empirically. Table 1 shows the summary of BD-rate[10] gain for standard test sequences with various resolutions. Basically, sequences with higher resolution gain more, since our proposed ASBH is effective for TB larger than 4x4. And high resolution sequences have more chance of using large TB in the coding process. In our experiments, our proposed ASBH outperforms the anchor for all tested sequences. It should be noted that the advanced tools in anchor HM9.0 have already been thoroughly studied in JCT-VC. It's not easy to gain over it, especially with very trivial complexity added at the decoder.

Table 1. Simulation results under Intra-Main configuration

Sequence	All Intra HE		
	Y(%)	U(%)	V(%)
Nebuta_4K	-0.20	-0.24	-0.24
SteamLocomotive_4K	-0.18	-0.31	-0.27
Kimono_1080P	-0.14	-0.31	-0.34
ParkScene_1080P	-0.13	-0.27	-0.29
BQTerrace_1080P	-0.14	-0.36	-0.35
RaceHorses_WVGA	-0.11	-0.22	-0.29
RaceHorses_WQVGA	-0.10	-0.21	-0.19
Johnny_720P	-0.10	-0.22	-0.30
Average	-0.14	-0.27	-0.29

5. CONCLUSION

In this paper, an additional sign bit hiding (ASBH) technique for H.265/HEVC is proposed. With the coordination to the existing MSBH in H.265/HEVC, our proposed scheme further exploits the potential of hiding information in TB coefficients by separating the coefficients into different levels, thus the conflicts between different hiding techniques in the quantized coefficients can be coordinated. Simulation results show that our proposed ASBH gains over H.265/HEVC averagely by -0.14%, -0.27%, -0.29% for Y, U and V respectively.

6. REFERENCES

[1] G.J. Sullivan, J.R. Ohm, et. al., "Overview of the High Efficient Video Coding (HEVC) Standard," *IEEE Trans. Circuits Syst. Video Tech.*, vol. 22, no. 12, pp. 1648-1667, Dec. 2012.

[2] J. Sole, R. Joshi, et. al., "Transform Coefficient Coding in HEVC," *IEEE Trans. Circuits Syst. Video Tech.*, vol. 22, no. 12, pp. 1765-1777, Dec. 2012.

[3] J.-M. Thiesse, J. Jung, et. al., "Rate Distortion Data Hiding of Motion Vector Competition Information in Chroma and Luma Samples for Video Compression," *IEEE Trans. Circuits Syst. Video Tech.*, vol. 21, no. 6, pp. 729-741, Jun. 2011.

[4] J.-M. Thiesse, J. Jung, et. al., "Data hiding of intra prediction information in chroma samples for video compression," *Proc, IEEE Conf. on Image Process. (ICIP)*, Sept. 2010.

[5] R. Cohen, S. Rane, et. al., "Low complexity Embedding of Information in Transform Coefficients," *JCTVC-E428*, JCT-VC meeting, Mar. 2011.

[6] H. Wang, M. Liu, et. al., "Region based motion vector prediction using data hiding and decoder side reasoning," *Visual Communications and Image Processing (VCIP)*, Nov. 2011.

[7] X. Yu, J. Wang, et. al., "Multiple Sign Bits Hiding," *JCTVC-H0481*, JCT-VC meeting, Feb. 2012.

[8] M. Karczewicz, Y. Ye, et. al., "Rate distortion optimized quantization," *VCEG-AH21*, VCEG meeting, Jan. 2008.

[9] F. Bossen, "Common HM test conditions and software reference configurations," *JCTVC-K1100*, JCT-VC meeting, Jul. 2011.

[10] G. Bjontegaard, "Calculation of average psnr differences between rd curves," *VCEG-M33*, VCEG meeting, Apr. 2001.