

## INDUSTRIAL TECHNOLOGY ADVANCES

# Residual sign prediction in transform domain for next-generation video coding

ALEXEY FILIPPOV<sup>1</sup>, VASILY RUFITSKIY<sup>1</sup>, ALEXANDER KARABUTOV<sup>1</sup> AND JIANLE CHEN<sup>2</sup>

*In this paper, we present a technique that is known as Residual Sign Prediction in Transform Domain (TDRSP) and is aimed at increasing compression performance by reducing the bits overhead of residue sign. These signs are typically coded by entropy coders in bypass mode that results in the high cost of sign bins which require 1 bit per bin in a bitstream. TDRSP allows us to reduce this cost by predicting residue signs so that the probability of a guess is significantly higher than 50%. Hence, arithmetic coding with contexts becomes applicable to the signs. In contrast to Residual Sign Prediction (RSP) performed in spatial domain, TDRSP avoids switching between domains and carries out calculations completely in transform domain to efficiently decrease the computational complexity of RSP. Simulations performed on top of Versatile Video Coding test model reference software (VTM-1.0) in accordance with the Joint Video Experts Team common test conditions show that more than 2.0% and up to 1.8% of the Bjøntegaard Delta rate can be achieved for All Intra and Random Access configurations, respectively.*

**Keywords:** Video coding, Residual sign prediction, Transform domain

Received 14 September 2018; Revised 13 April 2019; Accepted 15 April 2019

## I. INTRODUCTION

Entropy coding is one of the fundamental processing stages aimed at increasing the compression performance of video codecs. Currently, ISO/IEC MPEG and ITU-T VCEG standards such as AVC/H.264 (Main profile and higher) as well as HEVC/H.265 use Context-Adaptive Binary Arithmetic Codec (CABAC) as an entropy coding engine. CABAC typically provides bit-rate savings of 9–14% relative to Context-Adaptive Variable-Length Coding (CAVLC) at the same objective video quality [1]. It is worth noting that transform coefficients significantly contribute to bit-rate of video streams produced by AVC/H.264 and HEVC/H.265 encoders. Thus, to reduce the bit-rate related to transform coefficient coding, probabilities (frequencies) of each relevant syntax element should be accurately modeled to provide, in turn, their estimations to the CABAC arithmetic coding engine for achieving higher compression ratios. In this case, binary strings (bins), which have equiprobable states, are challenging for CABAC since they are coded in bypass mode and, hence, cannot take advantage of arithmetic coding. Among different syntax elements belonging

to quantized transform coefficients, their signs are represented as bins with equiprobable values and, therefore, require 1 bit per sign that increases a bit-rate of produced video streams.

One of the mechanisms that can realize it is sign data hiding (SDH) initially proposed in [2] and improved in [3]. The last modification was included in the HM (HEVC Model) software starting from version 6.0 and finally adopted into the HEVC/H.265 standard. Briefly, the basic idea behind SDH is to remove some of “expensive” residual signs at the cost of distortion increase caused by hiding the sign values in the sums of quantized transform coefficients within either a transform unit (TU) or a coefficient group (CG) of  $4 \times 4$  size in transform domain. To derive the sign values, parity of these sums is checked at decoder side. This approach is well substantiated from the view point of the RD theory and enables low-complexity implementations on encoder and especially decoder.

However, it does not take into account specific properties of block-based partitioning mechanism that is typical for hybrid video coding used in particular by standards of the H.26x series. As known, transform coding is used to reduce spatial correlation of residual signal within a block. However, pixels on the boundary between blocks have high correlation that in fact is a source of redundancy not yet utilized directly even by the latest video coding standard such as HEVC/H.265. This paper addresses this problem by predicting signs of residues exploiting property of spatial correlation between adjacent samples of neighboring blocks and calculating a cost function based on estimating

<sup>1</sup>Huawei Technologies, Moscow Research Center, Media Algorithms laboratory, 17/2, Krylatskaya str., Moscow 121614, Russia<sup>2</sup>Huawei Technologies, 2330 Central Expy, Santa Clara, CA 95050, USA**Corresponding author:**

J. Chen,

Email: [jianle.chen@huawei.com](mailto:jianle.chen@huawei.com)

discontinuities over block boundaries. In particular, we demonstrate that the cost function can be efficiently calculated only in transform domain, i.e. avoiding switching between these domains for the sake of reducing computational complexity.

This paper is based on the results presented in the Joint Video Experts Team (JVET) of ITU-T VCEG (Visual Coding Experts Group) and ISO/IEC MPEG (Moving Picture Experts Group) as proposals for the emerging Versatile Video Coding (VVC/H.266) standard. This technique known as Residual Sign Prediction in Transform Domain (TDRSP) was part of the joint Huawei, Samsung, GoPro, and HiSilicon responses [4, 5] to the Joint Call for Proposals (CfP) on Video Compression with Capability beyond HEVC issued by JVET in October 2017 [6]. In addition, TDRSP was and is being studied in the JVET Core Experiment on quantization and coefficient coding [7, 8].

## II. OVERVIEW

### A) Transform coefficients prediction techniques

As mentioned above, an inherent fundamental problem of block-based approach to still picture and video coding is blockiness referred to as blocking artifacts. In fact, they are discontinuities along block boundaries. Being perceptually annoying, they adversely affect subjective quality, especially, for low bit-rates when they become extraordinarily visible. To remedy this issue, post-processing or in-loop deblocking filters are used [9, 10]. As shown in Fig. 1, these filters usually take several (e.g., four) samples closest to a block on both sides of its boundary to analyze the drop between blocks and to make a decision to apply a filter to the ultimate samples or not. If decided to perform deblocking, its strength should be selected as well before applying the filter to the samples.

Deblocking filter can both provide significant objective gain and considerably improve the subjective quality of

lossy compressed pictures [9, 10]. However, it impacts a very limited number of samples within large blocks such as  $64 \times 64$  or  $32 \times 16$ . In particular, samples in the block center are usually not affected by deblocking filters.

On the other hand, before blocking, artifacts are filtered out, these discontinuities can be utilized to reduce rate of bit-streams generated by a codec. In particular, DC coefficients can be estimated by exploiting this interblock redundancy using different criteria that measure differences between pixels along block boundaries [11–13]. As these criteria, the following metrics can be used:

- minimum edge difference (MED) defined as  $MED = \sum_i \sum_j d_{1,ij}^2 + \sum_i \sum_j d_{2,ij}^2$  ( $d_{1,ij}$  and  $d_{2,ij}$  are explained in Fig. 2) and applied either locally (i.e. a given block) [11] or to a whole image [12];
- maximum a posterior (MAP) estimation with the Huber minimax function, which preserves the discontinuity of the pixels values across the high activity areas and the edge locations minimizing discontinuities on all boundaries of image [13].

Since the only prediction mechanism used in JPEG is applied to DC coefficients that are coded as differences of a DC coefficient of a given block and the DC coefficient of the previous one [14], these techniques were originally applied to and tested using still images compressed by JPEG-compliant codec [12].

The next step was to use this paradigm for increasing compression efficiency of video coding. So, prediction of DC and AC coefficients of intra-coded blocks was proposed for MPEG-1, MPEG-2, and H.263 bit-streams in [15]. The DC coefficient for a current block is predicted by the DC value of its either left or above neighboring block based on the comparison of horizontal and vertical gradients. A similar mechanism is applied to AC coefficients of top row and left column. Although a straightforward prediction mechanism was used, this method demonstrated over 26% improvement in coding efficiency of CIF resolution for

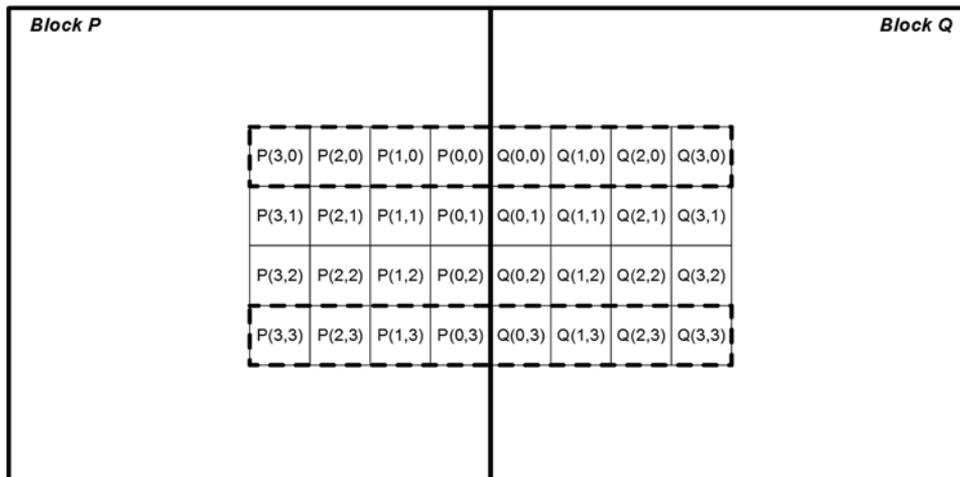


Fig. 1. Deblocking filtering.

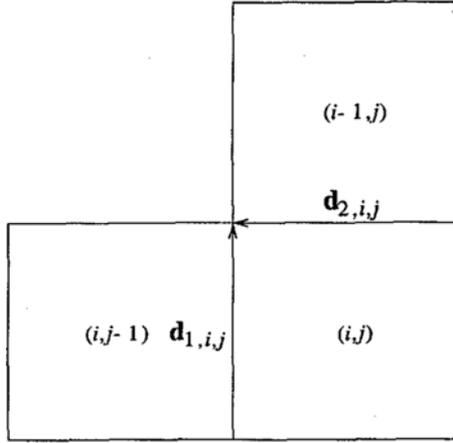


Fig. 2. Definitions of  $d_{1,i,j}$  and  $d_{2,i,j}$  [13].

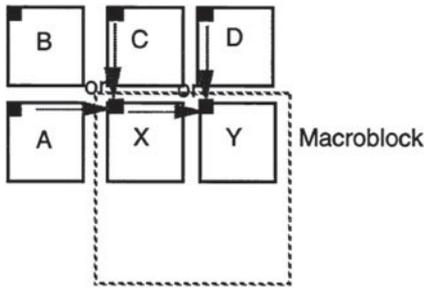


Fig. 3. Previous neighboring blocks used in improved DC prediction [15].

intra-coded video object planes as compared with H.263 (Fig. 3).

Further refinement of prediction mechanisms for DCT coefficients was presented in [16–18] where it was shown that for the vertical boundary of blocks, the DC coefficient of block  $k + 1$  was expressed through the DC coefficient of a left block and the differences of AC coefficients of these two blocks as follows:

$$c_{1,1}^{(k+1)} \approx \frac{N^2}{M} \sum_{j=0}^M (\rho_{0,j}^{(k)} - \rho_{0,j}^{(k+1)}) + c_{1,1}^{(k)},$$

where  $c_{1,1}$  is DC coefficient,  $\rho_{i,j}$  is AC part of pixel values.

A similar method of using boundary disparity and AC coefficients for this purpose was presented in [17]. This technique computes the difference of the sum of pixels of two boundary columns (or rows), one belonging to the current block and the other to a previous block, and then manipulates it in the direct cosine transform (DCT) domain so that the average of the coded differences for the whole image is near zero [17]. The experimental results reportedly demonstrated that this technique reduces the average JPEG DC residual by about 75% for images compressed at the default quality level [17] (Fig. 4).

Another view on the problem of predicting DCT coefficients was presented by H. Kondo *et al* that proposed to predict signs of DCT coefficients rather than their magnitudes [19]. In this case, Laplacian operator was used as a smoothness cost function that allowed the authors to

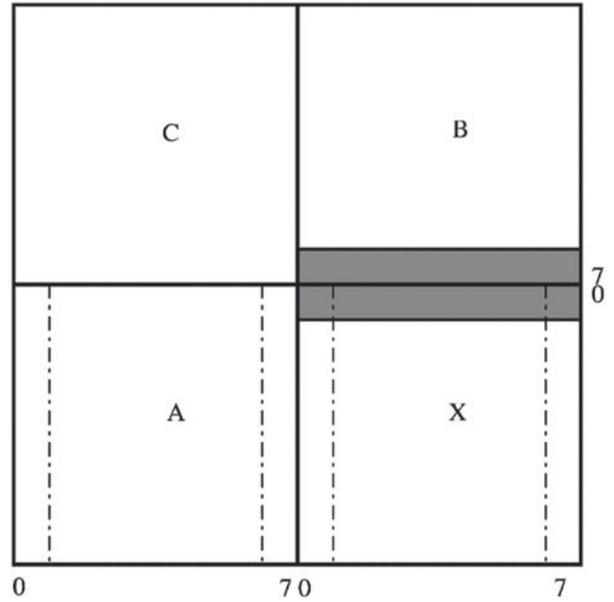


Fig. 4. Exploring inter-row/column redundancy for dc [17].

decrease bit-rate of JPEG-coded pictures by 0.1–0.2 bit/pixel on average [19]. In [20], a similar approach was described where an entropy-coded flag indicates whether a sign is correctly predicted or not. It resulted in increasing compression ratio of 3–9% that corresponds to up to 0.5 dB improvement in PSNR [20].

Estimation of the difference between neighboring samples of adjacent blocks could be performed with consideration of directionality, i.e. samples to calculate the differences for a boundary may belong to a direction, which is non-orthogonal to the boundary. In [16], values of differences are derived from pairs of samples in such a manner that samples within these pairs could have diagonal adjacency. Particularly, three sets of pixel pairs corresponding to one orthogonal-to-boundary direction and two diagonal directions are considered. The smoothest direction is selected in accordance with the minimum value of the mean squared error. As shown in Fig. 5, three sets of pixel pairs in two adjacent columns (rows) are considered that correspond to horizontal (vertical) and two diagonal directions.

According to experimental results of [16], it is reported that by using this approach the quality of still images compressed by JPEG could be increased up to 1.1 dB.

### III. SIGN PREDICTION IN SPATIAL DOMAIN

#### A) RSP technique in spatial domain

The technique initially proposed in [21] estimates the signs of a block from pixels in neighboring blocks and encodes the difference (o: same, 1: not same) between estimated signs and true signs by CABAC. If the signs are well estimated, the difference tends to be “o”, and they are efficiently

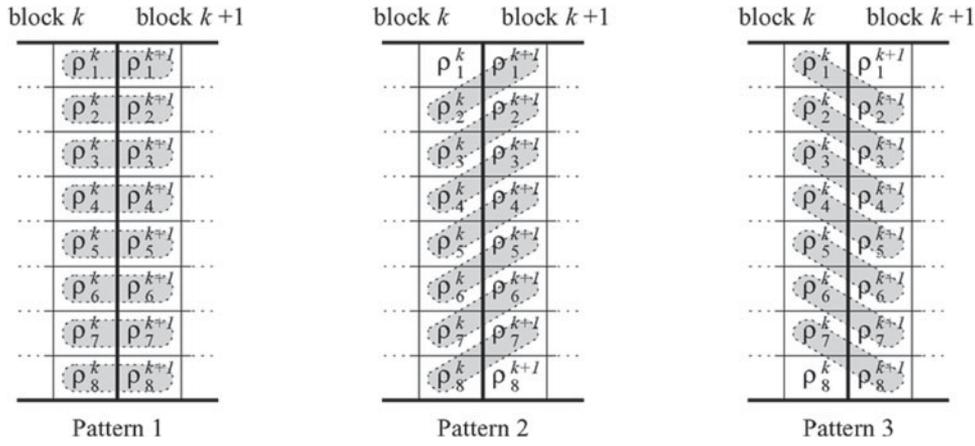


Fig. 5. Three patterns over a block boundary for two horizontally neighboring blocks [16].

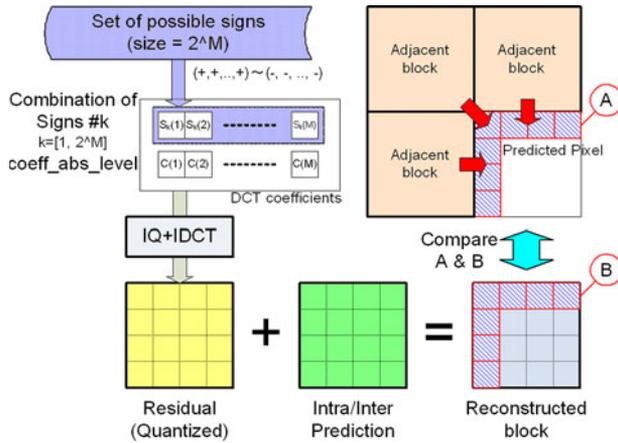


Fig. 6. Overview of sign estimation [21].

entropy-coded by CABAC. The concept of this tool is illustrated in Fig. 6.

In general, there is a high correlation among pixels at the boundary of a current block and those of neighboring blocks. The technique proposed in [21] predicts the signs of DCT coefficients of the current block using this property.

When there are  $N$  non-zero coefficients in the current block, the number of possible combinations of these signs is  $2^N$ . The proposed technique compares reconstructed pixels at the upper and left boundaries of the current block using every combination of signs with pixels extrapolated from neighboring blocks. The combination of signs which minimizes the square error is defined as estimated signs.

A comparison of the method proposed in [21] with the conventional H.264/AVC sign coding is shown in Fig. 7.

The steps of the technique in [21] are given in Table 1 where syntax element names are typed in bold.

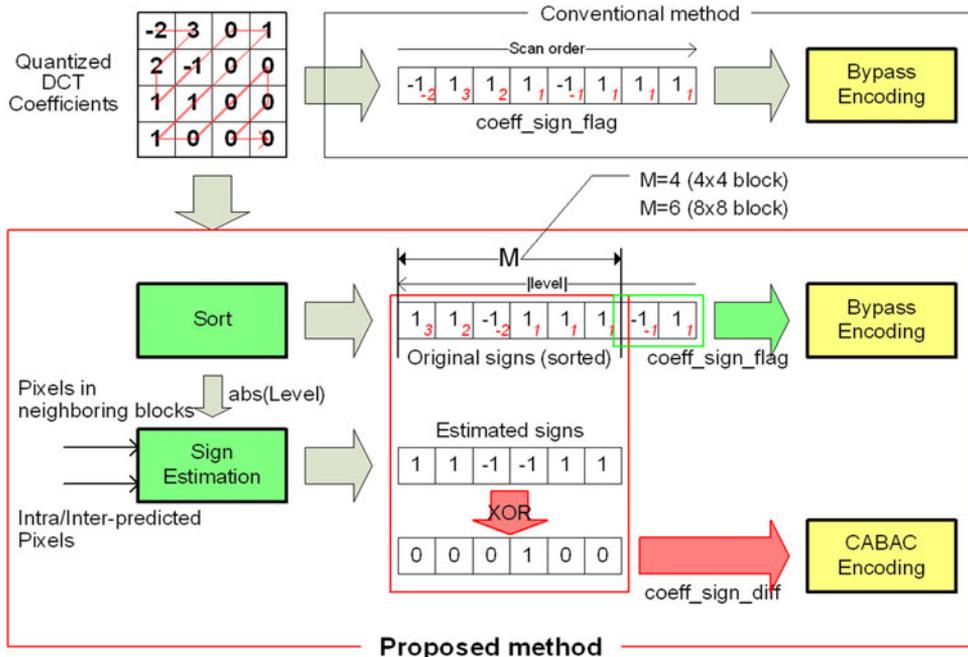
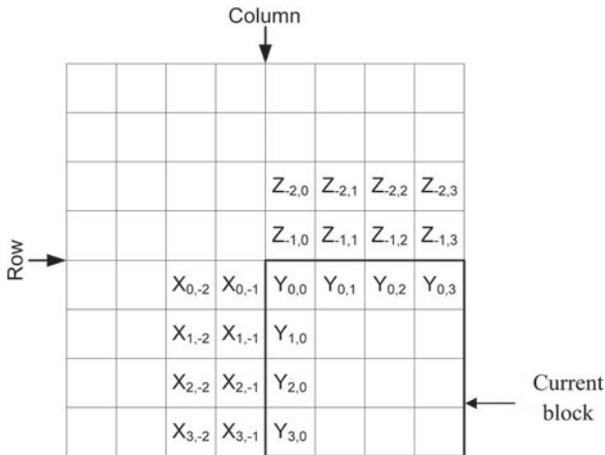


Fig. 7. Comparison of conventional sign coding with the technique proposed in [21].

**Table 1.** Steps of technique proposed in [21].

1	Sorting of coefficients	Coefficients are sorted by their absolute value of level
2	Sign estimation	Signs of coefficients are estimated as illustrated in Fig. 6 Specifically, there is the limitation of maximum number of signs to be estimated. Up to first $M$ signs in the sorted coefficients are estimated, where $M$ is equal to 4 for $4 \times 4$ block and 6 for $8 \times 8$ block. The reason for this limitation is that the estimation result of a sign of coefficient with small level tends to be wrong. The fast sign estimation method is introduced in order to reduce computational complexity.
3	Entropy coding	Exclusive OR operation is performed to the first $M^{\text{th}}$ signs of the sorted coefficients and their estimated values. The result, <b>coeff_sign_diff</b> is encoded using CABAC process. Other signs are encoded using the bypass process of CABAC (i.e. encoding process of <b>coeff_sign_flag</b> ).

**Fig. 8.** Pixels used to derive cost value in [22].

## B) Residual coefficient sign prediction

In October 2016, F. Henry and G. Claire proposed a sign prediction technique based on discontinuity measure between reconstructed and neighbor blocks [22]. One of the differences with [21] is that sign estimation is performed in [21] using L2 norm (sum of squared differences, SSD), while [23] uses L1 norm (sum of absolute differences).

Pixels used to derive a cost value are selected from neighboring blocks as shown in Fig. 8.

In spatial domain, cost function is defined in [21, 23] as:

$$F = \sum_{n=0}^{N-1} |2X_{n,-1} - X_{n,-2} - Y_{n,0}| + \sum_{m=0}^{M-1} |2Z_{-1,m} - Z_{-2,m} - Y_{0,m}|,$$

where  $N$  and  $M$  is height and width of the block.

**Table 2.** Hypotheses check for three coefficients by reusing templates.

Hypothesis	Calculation from template	Reused to calculate
$H_{+++}$	Prediction + $T_{001} + T_{010} + T_{100}$	$H_{++-}, H_{+-+}, H_{-++}$
$H_{+--}$	$H_{+++} - 2T_{001}$	not reused
$H_{-+-}$	$H_{+++} - 2T_{010}$	$H_{+--}$
$H_{--+}$	$H_{+++} - 2T_{001}$	not reused
$H_{-++}$	$H_{+++} - 2T_{100}$	$H_{--+}$
$H_{-+-}$	$H_{-++} - 2T_{001}$	not reused
$H_{--+}$	$H_{-++} - 2T_{010}$	$H_{---}$
$H_{---}$	$H_{--+} - 2T_{001}$	not reused

In [22], it was proposed to optimize hypotheses checking by using linear combination of pre-calculated templates to obtain a cost function value. Each template is a set of reconstructed pixels on the block boundary  $\{Y_{n,o}, Y_{o,m}\}$ , which is obtained by selecting a single coefficient belonging to a hypothesis and applying inverse transform to a matrix having all the coefficients equal to zero except the selected one. When calculating template, the absolute value of the selected single coefficient is utilized, and its sign is inferred in spatial domain by inverting signs of the template pixels  $\{Y_{n,o}, Y_{o,m}\}$  (Table 2).

Additionally, document [23] proposes a different method to encode predicted signs (step 3 in Table 1). In the reference software proposed together with [22], this modification was implemented by introducing two lists of predicted signs (modification of step 1 in Table 1).

Predicted signs belonging to these two lists are encoded with different CABAC contexts. The following rules are specified to populate these lists (see Fig. 9):

- First list is populated with signs of coefficients having magnitude greater than a predefined threshold  $T_1$ . The total number of signs in the first list is constrained by pre-defined value  $N$ ;
- If the number of signs in the first list  $n$  is lesser than  $N$ , the second list is being populated. The total number of signs in the second list is constrained by  $(N - n)$  so that the total number of signs in both lists does not exceed  $N$ . Coefficients populating the second list are sorted by their position in raster order, the magnitudes should not be greater than  $T_1$ .

The context to encode the sign is determined on whether it belongs to the first ( $L_1$ ) or to the second list ( $L_2$ ). Thus, this context determination mechanism essentially differs from step 3 in Table 1 that describes the single-context coding technique [21].

## IV. SIGN PREDICTION IN TRANSFORM DOMAIN

### A) Motivation

The main disadvantage of the RSP techniques described in Section II is that they require inverse transform to perform sign estimation, as the cost function  $F$  for selecting one of the signs sets determined by a hypothesis being checked is

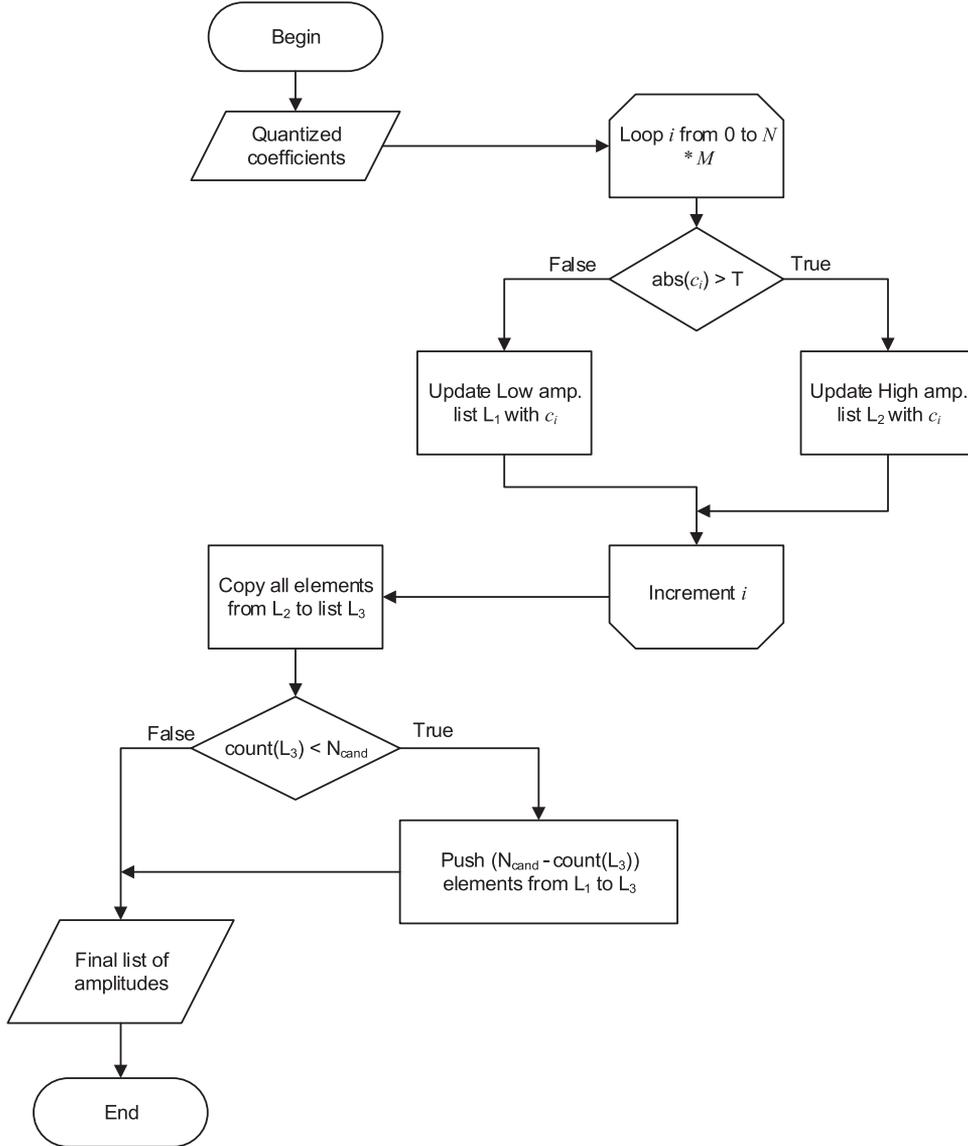


Fig. 9. Usage of two lists of coefficients for CABAC context determination.

calculated by using the differences of pixel values. Despite the presence of fast estimation methods, switching between transform and space domains is computationally expensive and is considered to be a significant drawback of these methods.

Instead of performing inverse transform of residual signal to obtain reconstructed boundary pixels, we proposed to perform 1D forward transform over the difference between pixels adjacent to the current block and pixels of prediction signal of the current block extrapolated to the area of the neighboring pixels. According to Parseval's identity, SSD calculated in transform domain for orthogonal transforms (such as DCT and DST) gives the same result as SSD calculated in spatial-domain.

Generally, the idea of predicting a sign value in transform domain is known (e.g., from [16–19]). This technique extends this approach to be applicable for the residual signal obtained after directional intra prediction. Additionally,

this technique is also applicable to the cases when adjacent blocks are coded using transforms of different types.

## B) Hypothesis checking in transform domain

As done in existing RSP method, this paper proposes to encode sign prediction error with a CABAC context model rather than in by-pass mode where each sign value is assumed to be equiprobable. An overview of the hypothesis checking process is shown in Fig. 10.

For the sake of simplicity, the only column of the adjacent block is shown in Fig. 10. Actually, the top row could also be used for hypothesis checking.

The basic idea underlying the concept of transform-domain sign prediction consists of estimating energy of discontinuity on the block boundary in transform domain without inverse transform of residual signal during hypothesis checking. For this purpose, the impact of

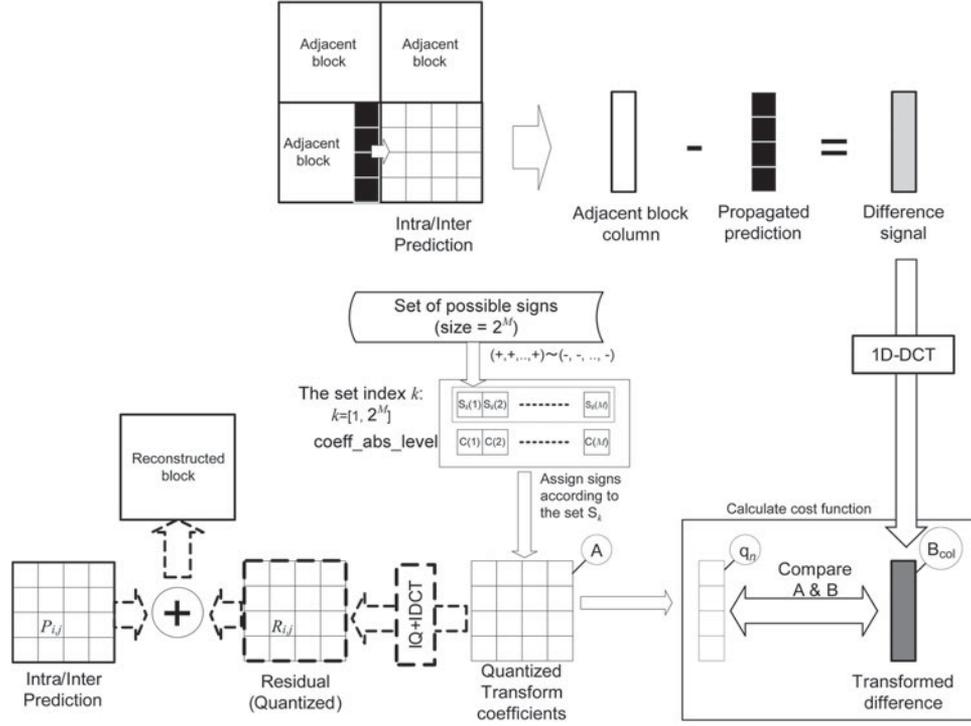


Fig. 10. Frequency-domain sign prediction concept.

prediction and residual signals on the energy of the discontinuity is estimated separately.

It is worth noting that estimating discontinuity in transform domain L1-norm will not be applicable since linear transform preserves energy of the signals but not their absolute values. Hence, rewriting cost estimation function proposed in [3, 4] to use L2-norm, i.e. a sum of squared differences (SSD), gives:

$$F = \sum_{n=0}^{N-1} (2X_{n,-1} - X_{n,-2} - Y_{n,o})^2 + \sum_{m=0}^{M-1} (2Z_{-1,m} - Z_{-2,m} - Y_{o,m})^2,$$

where  $N$  and  $M$  are height and width of the block, respectively.

Samples of reconstructed block could be represented as a decomposition of prediction and residual components:

$$Y_{i,j} = P_{i,j} + R_{i,j},$$

where  $P_{i,j}$  are predicted samples,  $R_{i,j}$  are residuals in spatial domain.

After substitution and rearrangement, residual signal in spatial domain can be expressed explicitly:

$$F = \sum_{n=0}^{N-1} ([2X_{n,-1} - X_{n,-2} - P_{n,o}] - R_{n,o})^2 + \sum_{m=0}^{M-1} ([2Z_{-1,m} - Z_{-2,m} - P_{o,m}] - R_{o,m})^2$$

Prediction component and neighboring samples are invariant to residual sign manipulation, so it is reasonable to use the following denotations:

$$T_n = [2X_{n,-1} - X_{n,-2} - P_{n,o}], \\ V_m = [2Z_{-1,m} - Z_{-2,m} - P_{o,m}], \\ Q_n = R_{n,o}, \quad \text{and} \quad O_m = R_{o,m}.$$

Applying Parseval's identity, cost function could be estimated in transform domain:

$$F = \sum_{n=0}^{N-1} (T_n - Q_n)^2 + \sum_{m=0}^{M-1} (V_m - O_m)^2 \\ \equiv \sum_{n=0}^{N-1} (t_n - q_n)^2 + \sum_{m=0}^{M-1} (v_m - o_m)^2,$$

where  $t_n = \text{Trans1D}(T_n)$ ,  $q_n = \text{Trans1D}(Q_n)$ ,  $v_m = \text{Trans1D}(V_m)$ ,  $o_m = \text{Trans1D}(O_m)$ ,  $\text{Trans1D}()$  is a 1D orthogonal transform.

Finally, we could calculate cost function, and, therefore, predict signs of quantized transform coefficients in transform domain. Moreover, there is no need to recalculate cost function value per sign combination completely. It is enough to just re-estimate its residual components, i.e.  $o_m$  and  $q_n$ .

### C) Harmonization with transforms and SBH

The concept of sign prediction is to estimate and utilize non-equality of the probabilities of the coefficients being positive or negative. However, this concept is inapplicable

to some of the blocks or coefficients because of interference with other techniques.

One of the interfering techniques is a SBH that infers the sign value of the first coefficient within a scan order of a coding group composed of 16 transform coefficients. By modifying the absolute values of quantized transform coefficients within the group, the parity of their sum is being controlled at the encoder side. The parity is interpreted by a decoder as a sign bit of the first coefficient within this group. However, SBH utilizes redundancy of a different kind than Residual Sign Prediction does. Specifically, SBH utilizes inefficiency of a scalar quantizer [24], while RSP (in both spatial and transform domains) estimates discontinuity with the neighboring blocks.

Thus, RSP could be harmonized with SBH by just ignoring the coefficients the signs of which are hidden by SBH. Since the nature of these two techniques is different, this way of harmonization does not introduce overlap of compression performance. Moreover, SBH is performed during

parsing process, while RSP is performed in decoding process (because it requires at least restored coefficient values), and therefore these techniques do not interfere and could be used jointly. In the next section, simulation results confirm that interference between these two techniques is small.

RSP techniques rely on the properties of transform that is applied to the residual signal. If performed in spatial domain, hypothesis check requires to perform inverse transforms that are computationally exhaustive in case secondary transform is used. Introduction of non-separable secondary transform (NSST) renders spatial-domain RSP to be impractical, and transform-domain RSP to be impossible. The latter technique utilizes the properties of Parseval's identity that does not hold for NSST. Hence, the most feasible way to harmonize an RSP technique with secondary transforms is to skip blocks to which secondary transforms are applied.

Enhanced multiple transform (EMT) is a less RSP-abusing technique, since a selected transform is applied to a

Table 3. TDRSP performance over VTM1.0.

	All Intra Main 10			Random Access Main 10			Low delay B Main 10			Low delay P Main 10		
	Y (%)	U (%)	V (%)	Y (%)	U (%)	V (%)	Y (%)	U (%)	V (%)	Y (%)	U (%)	V (%)
Tango 2	-1.99	-1.67	-1.32	-1.44	-1.21	-0.69						
Food market 4	-2.15	-1.41	-1.37	-1.27	-1.12	-1.13						
Camp fire	-0.39	-0.53	-0.71	-0.61	-0.49	-0.41						
Cat Robot 1	-1.49	-1.29	-1.16	-1.19	-1.01	-1.04						
Daylight road 2	-1.34	-1.05	-1.27	-0.97	-0.44	-0.51						
Park running 3	-0.77	-0.86	-1.00	-0.84	-0.78	-0.88						
Market place	-1.19	-0.86	-0.64	-0.92	-0.74	-0.73	-0.69	-0.12	-0.51	-0.83	-0.26	-0.40
Ritual dance	-2.05	-1.54	-1.37	-1.77	-1.29	-1.15	-1.51	-1.34	-1.00	-1.62	-1.29	-1.30
Cactus	-1.27	-1.00	-1.00	-1.23	-0.67	-1.10	-0.95	-0.15	-0.38	-1.11	-0.34	-0.56
Basketball drive	-1.18	-1.05	-0.92	-1.12	-0.65	-0.55	-0.97	-0.11	-0.70	-1.02	-0.60	-0.59
BQ terrace	-0.83	-0.70	-0.71	-0.61	-0.40	-0.89	-0.16	-0.07	0.36	-0.22	-0.02	-0.47
Basketball drill	-0.89	-0.63	-1.07	-1.23	-1.07	-0.85	-1.22	-0.69	-0.85	-1.41	-0.90	-0.95
BQ mall	-1.21	-1.03	-0.95	-0.89	-0.41	-1.34	-0.55	-0.50	0.00	-0.61	-0.38	-0.73
Party scene	-0.46	-0.53	-0.51	-0.46	-0.54	-0.46	-0.25	-0.26	0.13	-0.26	-0.50	0.03
Race horses	-0.76	-0.64	-0.83	-0.76	-0.37	-0.67	-0.61	-0.09	-0.20	-0.52	-0.45	-0.52
BasketballPass	-1.02	-0.62	-1.11	-0.98	-0.45	-0.79	-0.70	-0.11	-0.71	-0.77	-0.90	-0.51
BQSquare	-0.42	-0.52	-0.64	-0.12	-0.31	-1.64	-0.08	-0.46	0.48	-0.12	0.31	0.13
Blowing bubbles	-0.52	-0.33	-0.64	-0.39	-0.64	-0.06	-0.40	-0.22	-0.29	-0.29	-0.31	0.14
Race horses	-0.93	-0.68	-0.60	-0.79	-0.39	-0.75	-0.62	-0.36	0.38	-0.64	-0.27	-0.18
FourPeople	-1.88	-1.24	-1.41				-0.81	-2.07	-1.73	-0.97	-0.75	-1.26
Johnny	-1.74	-1.32	-1.40				-0.39	0.17	2.02	-0.56	-1.30	0.82
Kristen and Sara	-1.81	-1.36	-1.34				-0.35	0.53	-0.32	-0.33	-0.33	0.75
Basketball drill text	-0.79	-0.76	-0.84	-1.05	-0.69	-0.69	-1.01	-0.27	-0.70	-1.06	-1.13	-0.67
China speed	-0.38	-0.31	-0.30	-0.34	0.06	-0.23	-0.15	-0.36	-0.50	-0.24	-0.34	-0.19
Slide editing	-0.04	0.02	-0.07	0.04	0.00	-0.02	-0.08	-0.37	-0.19	-0.50	-0.88	-0.83
Slide show	-0.75	-0.82	-0.89	-0.71	-0.05	-0.63	-1.02	-1.74	0.20	-0.54	-0.57	-0.58
Class A1	-1.51	-1.20	-1.13	-1.11	-0.94	-0.75						
Class A2	-1.20	-1.07	-1.14	-1.00	-0.75	-0.81						
Class B	-1.30	-1.03	-0.93	-1.13	-0.75	-0.89	-0.86	-0.36	-0.45	-0.96	-0.50	-0.67
Class C	-0.83	-0.71	-0.84	-0.83	-0.60	-0.83	-0.66	-0.39	-0.23	-0.70	-0.56	-0.54
Class E	-1.81	-1.31	-1.38				-0.52	-0.46	-0.01	-0.62	-0.79	0.11
<b>Overall</b>	<b>-1.30</b>	<b>-1.04</b>	<b>-1.05</b>	<b>-1.02</b>	<b>-0.75</b>	<b>-0.83</b>	<b>-0.70</b>	<b>-0.39</b>	<b>-0.27</b>	<b>-0.79</b>	<b>-0.59</b>	<b>-0.43</b>
<b>Overall complexity</b>												
<b>Enc</b>		<b>119</b>			<b>109</b>			<b>109</b>			<b>110</b>	
<b>Dec</b>		<b>151</b>			<b>118</b>			<b>120</b>			<b>120</b>	
Class D	-0.72	-0.53	-0.75	-0.57	-0.45	-0.81	-0.45	-0.29	-0.04	-0.46	-0.29	-0.10
Class F (optional)	-0.49	-0.47	-0.52	-0.51	-0.17	-0.39	-0.56	-0.69	-0.30	-0.59	-0.73	-0.57

residual signal only once (just as in conventional case), and the transform is still orthogonal and separable. However, in the case of TDRSP, the difference in transform type in horizontal and vertical directions require special handling when performing forward 1D-transform of the neighboring area. Specifically, the transform type should be the same for adjacent column of the neighbor block and vertical transform or the residual signal, as well as for the adjacent row of the upper block and horizontal transform of the residual signal. Evidently, EMT signaling should precede transform coefficients coding in the parsing process.

## D) Impact on hardware complexity

TDRSP is computationally easier than any of spatial-domain RSP method since it eliminates inverse transforms that are a bottleneck of spatial-domain methods. The only significant computational overhead of TDRSP consists in applying two additional 1D forward transforms, for a column and a row of a block, respectively. Another drawback

of TDRSP is in its incompatibility with secondary transforms that leads to significant performance degradation on intra-slices where secondary transforms are enabled.

The major bottleneck for both spatial and transform-domain methods of RSP is in their inherent dependency between inverse transform and block prediction (both intra and inter). In fact, inverse transform process requires prediction signal of the neighboring block to infer signs, which introduces additional delay and makes it almost impossible to perform inverse transforms in parallel. Removal of dependency from the neighboring block (e.g. by modifying cost function to minimize residual energy on the boundary instead of discontinuity estimation) can eliminate the source of redundancy for RSP and TDRSP. However, this will lead to a total loss of coding performance.

## E) Experimental results

TDRSP technique has been implemented and tested in the test model reference software for next-generation video

**Table 4.** TDRSP performance over VTm1.0, TDRSP applied only to intra-coded blocks.

	All Intra Main 10			Random Access Main 10			Low delay B Main 10			Low delay P Main 10		
	Y (%)	U (%)	V (%)	Y (%)	U (%)	V (%)	Y (%)	U (%)	V (%)	Y (%)	U (%)	V (%)
Tango 2	-1.99	-1.67	-1.32	-0.75	-0.86	-0.41						
Food market 4	-2.15	-1.41	-1.37	-0.71	-0.70	-0.72						
Camp fire	-0.39	-0.53	-0.71	-0.35	-0.31	-0.25						
Cat robot 1	-1.49	-1.29	-1.16	-0.64	-0.45	-0.54						
Daylight road 2	-1.34	-1.05	-1.27	-0.64	-0.20	-0.40						
Park running 3	-0.77	-0.86	-1.00	-0.31	-0.29	-0.36						
Market place	-1.19	-0.86	-0.64	-0.47	-0.31	-0.41	-0.19	-0.01	-0.09	-0.36	-0.10	-0.33
Ritual dance	-2.05	-1.54	-1.37	-1.05	-0.60	-0.53	-0.53	-0.58	-0.62	-0.67	-0.65	-0.84
Cactus	-1.27	-1.00	-1.00	-0.66	-0.39	-0.71	-0.21	0.23	0.12	-0.29	0.08	0.01
Basketball drive	-1.18	-1.05	-0.92	-0.62	-0.37	-0.31	-0.20	0.11	-0.59	-0.36	-0.38	-0.38
BQ terrace	-0.83	-0.70	-0.71	-0.47	-0.23	-0.53	-0.05	-0.51	1.38	-0.08	1.07	1.68
Basketball drill	-0.89	-0.63	-1.07	-0.57	-0.59	-0.47	-0.12	-0.11	0.14	-0.33	-0.62	0.47
BQ mall	-1.21	-1.03	-0.95	-0.53	0.04	-0.77	-0.11	-0.50	0.01	-0.17	-0.22	-0.63
Party scene	-0.46	-0.53	-0.51	-0.25	-0.23	-0.19	-0.10	-0.08	0.06	-0.03	-0.60	0.13
Race horses	-0.76	-0.64	-0.83	-0.39	-0.08	-0.32	-0.15	0.00	0.01	-0.17	-0.26	-0.33
Basketball pass	-1.02	-0.62	-1.11	-0.50	-0.33	-0.66	-0.07	-0.19	-0.48	-0.25	-0.77	-0.40
BQ square	-0.42	-0.52	-0.64	-0.17	0.01	-1.86	-0.02	-0.63	-0.29	0.00	-0.25	0.90
Blowing bubbles	-0.52	-0.33	-0.64	-0.22	-0.16	-0.01	-0.09	-0.07	0.16	0.01	-0.49	0.46
Race horses	-0.93	-0.68	-0.60	-0.34	-0.16	-0.45	-0.17	0.01	0.45	-0.09	0.29	-0.28
Four people	-1.88	-1.24	-1.41				-0.21	-1.60	-1.30	-0.37	-0.13	-0.46
Johnny	-1.74	-1.32	-1.40				0.01	0.93	1.48	-0.32	-1.11	0.79
Kristen and Sara	-1.81	-1.36	-1.34				0.11	0.75	-0.07	-0.07	0.21	0.21
Basketball drill text	-0.79	-0.76	-0.84	-0.45	-0.47	-0.31	-0.15	-0.48	0.02	-0.29	-0.53	0.10
China speed	-0.38	-0.31	-0.30	-0.20	0.00	-0.30	-0.02	-0.14	-0.44	-0.02	-0.24	-0.36
Slide editing	-0.04	0.02	-0.07	0.08	0.01	-0.01	-0.21	-0.54	-0.36	-0.45	-0.72	-0.76
Slide show	-0.75	-0.82	-0.89	-0.61	0.02	-0.56	-0.55	-0.03	-1.62	-0.03	-1.71	-0.54
Class A1	-1.51	-1.20	-1.13	-0.60	-0.62	-0.46						
Class A2	-1.20	-1.07	-1.14	-0.53	-0.31	-0.43						
Class B	-1.30	-1.03	-0.93	-0.65	-0.38	-0.50	-0.24	-0.15	0.04	-0.35	0.00	0.03
Class C	-0.83	-0.71	-0.84	-0.43	-0.21	-0.44	-0.12	-0.17	0.06	-0.17	-0.43	-0.09
Class E	-1.81	-1.31	-1.38				-0.03	0.03	0.04	-0.25	-0.35	0.18
Overall	<b>-1.30</b>	<b>-1.04</b>	<b>-1.05</b>	<b>-0.56</b>	<b>-0.37</b>	<b>-0.46</b>	<b>-0.15</b>	<b>-0.11</b>	<b>0.04</b>	<b>-0.27</b>	<b>-0.23</b>	<b>0.03</b>
<b>Overall complexity</b>												
<b>Enc</b>		<b>119</b>			<b>105</b>			<b>104</b>			<b>105</b>	
<b>Dec</b>		<b>151</b>			<b>112</b>			<b>109</b>			<b>111</b>	
<b>Class D</b>	-0.72	-0.53	-0.75	-0.31	-0.16	-0.74	-0.09	-0.22	-0.04	-0.08	-0.30	0.17
<b>Class F (optional)</b>	-0.49	-0.47	-0.52	-0.29	-0.11	-0.29	-0.24	-0.30	-0.60	-0.20	-0.80	-0.39

Table 5. TDRSP performance over BMS1.0.

	All Intra Main 10			Random Access Main 10			Low delay B Main 10			Low delay P Main 10		
	Y (%)	U (%)	V (%)	Y (%)	U (%)	V (%)	Y (%)	U (%)	V (%)	Y (%)	U (%)	V (%)
Tango 2	-0.68	-0.56	-0.56	-0.94	-0.50	-0.49						
Food market 4	-0.75	-0.24	-0.43	-0.74	-0.51	-0.51						
Campfire	-0.22	-0.18	-0.28	-0.34	-0.28	-0.37						
Cat robot 1	-0.48	-0.47	-0.45	-0.75	-0.82	-0.50						
Daylight road 2	-0.51	-0.17	-0.50	-0.53	-0.07	-0.26						
Park running 3	-0.28	-0.24	-0.30	-0.58	-0.69	-0.79						
Market place	-0.36	-0.24	-0.45	-0.62	-0.31	-0.46	-0.49	-0.60	-0.38	-0.45	-0.73	-0.74
Ritual dance	-0.67	-0.38	-0.42	-1.05	-0.75	-0.81	-1.06	-0.73	-0.70	-1.10	-0.96	-0.58
Cactus	-0.35	-0.34	-0.35	-0.72	-0.33	-0.59	-0.86	0.14	-0.16	-0.85	0.04	-0.54
Basketball drive	-0.46	-0.19	-0.29	-0.68	-0.71	-0.58	-0.78	-0.28	-0.37	-0.82	-0.13	-0.39
BQ terrace	-0.29	-0.34	-0.23	-0.21	0.36	-0.44	-0.10	-0.83	1.40	-0.19	-0.13	-0.47
Basketball drill	-0.20	-0.31	-0.05	-0.78	-0.59	-0.54	-1.34	-0.14	-1.07	-1.24	-0.61	-1.25
BQ mall	-0.49	-0.46	-0.47	-0.64	-0.57	-0.93	-0.47	-0.61	-0.21	-0.48	-0.18	-0.57
Party scene	-0.15	-0.18	-0.26	-0.26	-0.55	-0.78	-0.22	-0.25	-0.81	-0.29	-0.61	-0.20
Race horses	-0.21	-0.28	-0.29	-0.37	0.06	-0.66	-0.47	-0.13	0.07	-0.45	-0.79	-0.55
Basketball pass	-0.31	-0.11	-0.18	-0.57	-0.69	-0.79	-0.63	-0.99	-0.70	-0.71	-0.33	-0.61
BQ square	-0.16	-0.07	-0.10	-0.04	-0.07	-1.05	-0.05	1.86	-0.93	-0.01	0.67	0.71
Blowing bubbles	-0.15	-0.27	-0.27	-0.24	-0.34	0.05	-0.18	-0.24	0.12	-0.12	-1.02	-0.47
Race horses	-0.25	0.01	-0.16	-0.39	-0.53	-0.72	-0.54	-0.41	-0.27	-0.39	-0.19	-0.63
Four people	-0.70	-0.51	-0.55				-0.61	-0.28	0.96	-0.63	-0.70	-1.14
Johnny	-0.65	-0.43	-0.69				-0.60	-0.17	0.15	-0.15	0.12	-2.88
Kristen and Sara	-0.61	-0.45	-0.41				-0.30	-1.40	-0.49	-0.59	-1.80	-0.67
Basketball drill text	-0.24	-0.17	-0.17	-0.69	-0.50	-0.65	-0.78	-0.85	-0.40	-1.10	-1.06	-0.96
China speed	-0.22	-0.44	-0.22	-0.16	0.14	0.03	-0.23	-0.59	-0.59	-0.37	-0.18	-0.14
Slide editing	-0.04	-0.10	-0.11	0.02	-0.02	0.06	-0.17	-0.25	-0.35	0.41	0.60	0.59
Slide show	-0.46	0.04	-0.76	-0.64	-0.86	0.06	-0.50	-0.72	3.28	-0.41	-0.49	-0.42
Class A1	-0.55	-0.32	-0.42	-0.67	-0.43	-0.46						
Class A2	-0.42	-0.29	-0.42	-0.62	-0.52	-0.52						
Class B	-0.43	-0.30	-0.35	-0.66	-0.35	-0.57	-0.66	-0.46	-0.04	-0.68	-0.38	-0.54
Class C	-0.26	-0.31	-0.27	-0.51	-0.41	-0.73	-0.62	-0.28	-0.51	-0.61	-0.55	-0.64
Class E	-0.65	-0.47	-0.55				-0.50	-0.62	0.20	-0.46	-0.79	-1.56
<b>Overall</b>	<b>-0.45</b>	<b>-0.33</b>	<b>-0.39</b>	<b>-0.61</b>	<b>-0.42</b>	<b>-0.58</b>	<b>-0.61</b>	<b>-0.44</b>	<b>-0.14</b>	<b>-0.60</b>	<b>-0.54</b>	<b>-0.83</b>
<b>Overall complexity</b>												
<b>Enc</b>		<b>109</b>			<b>107</b>			<b>107</b>			<b>108</b>	
<b>Dec</b>		<b>121</b>			<b>110</b>			<b>112</b>			<b>112</b>	
Class D	-0.22	-0.11	-0.18	-0.31	-0.41	-0.63	-0.35	0.05	-0.44	-0.31	-0.22	-0.25
Class F (optional)	-0.24	-0.17	-0.31	-0.37	-0.31	-0.13	-0.42	-0.60	0.49	-0.37	-0.28	-0.23

coding standard VVC (VTM1.0) and the Benchmark Set (BMS1.0) [25]. The experiments are conducted by using the common test conditions for VVC development [26]. *Bjontegaard Delta (BD) rates* are used to measure coding performance improvement. As mentioned in [25], VTM1.0 only contain the basic coding block partitioning architecture, using a quadtree with a nested multi-type tree. All the additional coding tools, including EMT and NSST, are included in the BMS1.0 software.

Coding performance results are given in Tables 3–6 [27]. As shown in Table 3, on top of VTM1.0, the proposed TDRSP achieves 1.3, 1.0, and 0.7% luma BD rate on average, respectively, for All Intra, Random Access and Low Delay B configurations. The coding gain is up to 2.2 and 1.8%, respectively, for All Intra and Random Access configuration. Table 5 shows the coding gain of TDRSP on top of BMS1.0. The coding gain becomes smaller due to the conflict of TDRSP and NSST, as well as the interactions of TDRSP with other coding tools in BMS1.0.

As mentioned in the previous sub-section, the RSP method introduces additional dependency inverse transform

and block prediction (both intra and inter) and this feature may result in hardware implementation difficulty. To alleviate the impact of this dependency, we also tested the case that TDRSP only applied to intra-coded block. Tables 4 and 6 provide the coding performance for the case when TDRSP is only applied to intra-coded block.

Table 7 gives the estimation of interference between SBH and TDRSP.

Table 8 gives the per-sequence estimation of probabilities of correct sign prediction using TDRSP method. Estimation is performed separately for each EMT case, i.e. for each combination of horizontal and vertical transform types. It is shown that the estimation of sign value is quite accurate for the coefficients with large magnitude.

## V. CONCLUSION

RSP methods demonstrate significant and consistent coding performance on various content, up to 2% of BD-rate gain on some sequences. However, their inherent dependency

**Table 6.** TDRSP performance over BMS1.0, TDRSP applied only to intra-coded blocks.

	All Intra Main 10			Random Access Main 10			Low delay B Main 10			Low delay P Main 10		
	Y (%)	U (%)	V (%)	Y (%)	U (%)	V (%)	Y (%)	U (%)	V (%)	Y (%)	U (%)	V (%)
Tango 2	-0.68	-0.56	-0.56	-0.27	-0.23	-0.16						
Food market 4	-0.75	-0.24	-0.43	-0.26	-0.23	-0.28						
Campfire	-0.22	-0.18	-0.28	-0.18	-0.14	-0.23						
Cat robot 1	-0.48	-0.47	-0.45	-0.16	-0.31	0.09						
Daylight road 2	-0.51	-0.17	-0.50	-0.20	0.00	-0.06						
Park running 3	-0.28	-0.24	-0.30	-0.11	-0.10	-0.13						
Market place	-0.36	-0.24	-0.45	-0.20	0.04	-0.05	-0.09	-0.31	0.07	-0.09	-0.23	-0.48
Ritual dance	-0.67	-0.38	-0.42	-0.29	-0.21	-0.23	-0.15	-0.10	-0.22	-0.24	-0.10	-0.12
Cactus	-0.35	-0.34	-0.35	-0.20	-0.07	-0.23	-0.16	0.00	0.52	-0.18	-0.19	0.22
Basketball drive	-0.46	-0.19	-0.29	-0.26	-0.25	-0.25	-0.12	-0.15	0.04	-0.16	0.00	-0.27
BQ terrace	-0.29	-0.34	-0.23	-0.11	0.03	-0.15	-0.09	-1.06	-0.44	-0.07	0.28	-0.12
Basketball drill	-0.20	-0.31	-0.05	-0.18	-0.35	-0.39	-0.17	0.07	-0.41	-0.25	0.14	-1.01
BQ mall	-0.49	-0.46	-0.47	-0.19	-0.33	-0.87	-0.02	0.06	-0.16	-0.06	0.34	-0.23
Party scene	-0.15	-0.18	-0.26	-0.01	-0.31	-0.51	0.05	0.12	-0.17	-0.05	-0.21	-0.21
Race horses	-0.21	-0.28	-0.29	-0.04	0.37	0.01	-0.19	0.26	-0.06	-0.06	-0.18	-0.17
Basketball pass	-0.31	-0.11	-0.18	-0.27	-0.69	-0.39	-0.04	-0.85	-0.28	-0.06	0.01	-0.05
BQ square	-0.16	-0.07	-0.10	-0.12	-0.28	-1.28	0.15	-0.17	-2.65	-0.21	0.97	0.09
Blowing bubbles	-0.15	-0.27	-0.27	-0.06	-0.16	-0.12	-0.11	0.10	-0.38	0.11	-0.68	-0.52
Race horses	-0.25	0.01	-0.16	-0.03	-0.09	-0.14	-0.09	0.13	-0.08	-0.09	-0.34	-0.47
Four people	-0.70	-0.51	-0.55				-0.19	-0.33	1.38	-0.13	-0.24	-0.12
Johnny	-0.65	-0.43	-0.69				-0.37	-0.13	0.10	-0.20	0.03	-1.09
Kristen and Sara	-0.61	-0.45	-0.41				-0.23	-0.57	-0.80	-0.16	-1.92	-1.30
Basketball drill text	-0.24	-0.17	-0.17	-0.10	-0.03	-0.24	0.10	0.06	-0.19	-0.16	-0.48	-0.21
China speed	-0.22	-0.44	-0.22	-0.03	0.30	0.12	-0.04	-0.35	-0.56	-0.11	0.00	0.02
Slide editing	-0.04	-0.10	-0.11	0.03	-0.01	0.11	0.15	0.25	0.43	0.63	0.68	0.61
Slide show	-0.46	0.04	-0.76	-0.48	-0.72	0.27	0.01	0.64	1.79	-0.32	-1.30	-0.23
Class A1	-0.55	-0.32	-0.42	-0.24	-0.20	-0.22						
Class A2	-0.42	-0.29	-0.42	-0.16	-0.14	-0.03						
Class B	-0.43	-0.30	-0.35	-0.21	-0.09	-0.18	-0.12	-0.32	0.00	-0.15	-0.05	-0.15
Class C	-0.26	-0.31	-0.27	-0.10	-0.15	-0.44	-0.08	0.13	-0.20	-0.10	0.02	-0.40
Class E	-0.65	-0.47	-0.55				-0.26	-0.34	0.23	-0.16	-0.71	-0.83
<b>Overall</b>	<b>-0.45</b>	<b>-0.33</b>	<b>-0.39</b>	<b>-0.18</b>	<b>-0.14</b>	<b>-0.23</b>	<b>-0.15</b>	<b>-0.18</b>	<b>-0.01</b>	<b>-0.14</b>	<b>-0.19</b>	<b>-0.41</b>
<b>Overall complexity</b>												
<b>Enc</b>		<b>109</b>			<b>104</b>			<b>103</b>			<b>103</b>	
<b>Dec</b>		<b>122</b>			<b>107</b>			<b>106</b>			<b>107</b>	
Class D	-0.22	-0.11	-0.18	-0.12	-0.30	-0.48	-0.02	-0.20	-0.85	-0.06	-0.01	-0.24
Class F (optional)	-0.24	-0.17	-0.31	-0.15	-0.12	0.07	0.06	0.15	0.37	0.01	-0.28	0.05

**Table 7.** TDRSP performance over IFVC1.0, TDRSP applied only to intra-coded blocks.

	SBH			TDRSP			SBH + TDRSP		
	Random Access Main 10			Random Access Main 10			Random Access Main 10		
	Y (%)	U (%)	V (%)	Y (%)	U (%)	V (%)	Y (%)	U (%)	V (%)
Market place	-0.53	-0.62	-0.04	-0.99	-0.41	-0.71	-1.31	-1.01	-0.53
Ritual dance	-0.34	-1.88	-1.35	-2.05	-2.44	-1.49	-2.43	-2.65	-2.49
Cactus	-0.48	-1.49	-1.03	-1.29	-1.38	-1.06	-1.07	-1.31	-1.15
Basketball drive	-0.49	-0.76	-0.71	-0.47	-0.85	-0.97	-0.84	-1.25	-1.04
BQ terrace	0.23	-0.86	-2.29	-0.63	0.41	0.24	-0.72	-1.33	-0.79
Basketball drill	-0.32	-0.44	0.23	-1.10	-1.36	-0.01	-0.89	-1.37	-1.29
BQ mall	-0.30	-1.17	-0.15	-0.98	-0.88	-0.74	-1.05	-1.52	-1.29
Party scene	-0.61	-1.90	-1.35	-0.49	-0.90	-0.51	-0.75	-1.40	-1.41
Race horses	-0.48	-0.35	-1.46	-0.68	-0.83	-0.45	-0.91	-1.35	-1.03
Basketball pass	-0.90	0.23	0.85	-0.68	0.08	1.58	-0.77	-1.15	-1.38
BQ square	-0.28	-1.98	-1.54	-1.14	1.07	2.09	-1.27	-1.42	-1.02
Blowing bubbles	-0.50	-1.57	-0.53	-0.51	-1.87	0.98	-0.96	-1.40	-1.56
Race horses	-0.50	-1.21	-0.59	-0.74	-1.71	-0.88	-1.32	-1.69	-1.86
Class B	-0.32	-1.12	-1.08	-0.99	-0.93	-0.80	-1.27	-1.51	-1.20
Class C	-0.43	-0.96	-0.68	-0.41	-0.99	-0.43	-0.90	-1.41	-1.25
Class D	-0.55	-1.13	-0.45	-0.59	-0.61	0.94	-1.08	-1.42	-1.45

**Table 8.** Per-sequence estimation of probability of correct sign prediction TDRSP performance in Random access mode.

Class	Sequence	High amp,	Low amp.,								
		DCT2/ DCT2 (%)	DCT2/ DCT2 (%)	DCT8/ DCT8 (%)	DCT8/ DCT8 (%)	DCT2/ DST7 (%)	DCT2/ DST7 (%)	DST7/ DCT8 (%)	DST7/ DCT8 (%)	DST7/ DST7 (%)	DST7/ DST7 (%)
UHD	FoodMarket4	95.25	85.15	93.61	83.87	95.24	82.60	95.52	85.71	92.81	88.76
	CatRobot	90.32	81.67	83.99	75.96	85.80	77.28	84.55	76.21	88.13	77.13
	DaylightRoad2	86.77	77.76	80.43	71.84	82.21	72.19	82.38	72.13	84.32	74.48
	ParkRunning3	83.35	76.86	76.72	67.17	77.73	68.15	77.01	67.90	81.94	69.99
	CampfireParty2	81.57	77.28	74.45	68.85	76.74	68.75	75.44	67.72	79.96	68.13
HD	BQTerrace	79.12	60.66	75.03	56.89	76.21	59.51	76.37	57.27	75.18	59.66
	RitualDance	90.02	84.63	84.23	75.80	85.04	77.60	85.70	76.71	83.99	79.69
	MarketPlace	85.58	76.30	79.55	69.32	80.17	67.85	80.19	69.16	84.87	73.99
	BasketballDrive	84.86	76.16	78.31	67.49	78.00	67.10	81.18	69.14	81.12	74.76
	Cactus	85.39	76.69	76.93	67.49	79.23	68.40	79.61	67.85	81.08	71.37
C	BasketballDrill	88.68	79.04	84.59	70.89	84.52	74.48	85.20	75.00	83.02	80.00
	BQMall	79.83	69.94	76.15	63.50	77.00	66.41	77.86	63.52	78.07	64.74
	PartyScene	75.43	67.62	74.16	65.91	72.63	65.12	72.75	62.06	69.42	58.14
	RaceHorses	82.82	78.27	76.73	71.76	78.03	69.78	77.24	69.93	76.94	75.14
D	BasketballPass	77.92	64.35	77.15	54.86	73.38	57.14	79.40	72.04	77.52	76.19
	BQSquare	78.47	61.13	77.51	66.46	75.58	58.97	75.20	48.00	70.61	40.00
	BlowingBubbles	81.93	71.81	80.29	70.04	78.77	71.59	79.49	63.64	77.03	72.50
	RaceHorses	84.15	72.00	80.32	68.75	80.05	67.76	81.83	68.59	83.15	73.47

between inverse transform and prediction processes imply serious limitations on hardware design. These limitations prevent RSP and TDRSP from being widely adopted in state-of-the-art video coding.

The effect of the implied interdependency between inverse transform and prediction is less critical in software implementations thus making RSP feasible for them.

## STATEMENT OF INTEREST

The authors of this paper contributed to ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, document number JVET-K0044.

## REFERENCES

- [1] Marpe, D.; Schwarz, H.; Wiegand, T.: Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard. *IEEE Trans. Circuits Syst. Video Technol.*, 13 (7) (2003), 620–636.
- [2] Yu, X.; Wang, J.; He, D.; Martin-Cocher, G.; Campbell, S.: Multiple Sign Bits Hiding. Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC TC1/SC29/WG11, JCTVCH0481, 8th Meeting: San Jose, CA, USA, 1–10 February, 2012.
- [3] Clare, G.; Henry, F.; Jung, J.: Sign Data Hiding. in *Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, JCTVC-G271, 7th Meeting*, Geneva, CH, 21–30 November, 2011.
- [4] Alshin, A. *et al.*: Description of SDR, HDR and 360° video coding technology proposal by Samsung, Huawei, GoPro, and HiSilicon – mobile application scenario, in *Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, JVET-J0024, 10th Meeting*, San Diego, USA, 10–20 Apr. 2018.
- [5] Chen, H. *et al.*: Description of SDR, HDR and 360° video coding technology proposal by Huawei, GoPro, HiSilicon, and Samsung – general application scenario, in *Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, JVET-J0025, 10th Meeting*, San Diego, USA, 10–20 Apr. 2018.
- [6] Segall, A.; Baroncini, V.; Boyce, J.; Chen, J.; Suzuki, T.: Joint Call for Proposals on Video Compression with Capability beyond HEVC, in *Joint Video Exploration Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, JVET-H1002, 8th Meeting*, Macao, CN, 18–24 October 2017.
- [7] Schwarz, H.; Coban, M.: Description of Core Experiment 7 (CE 7): Quantization and coefficient coding, in *Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, JVET-J1027, 10th Meeting*, San Diego, USA, 10–20 Apr. 2018.
- [8] Schwarz, H.; Coban, M.; Auyeung, C.: Description of Core Experiment 7 (CE7): Quantization and coefficient coding, in *Joint Video Experts Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, JVET-K1027, 11th Meeting*, Ljubljana, SI, 10–18 July 2018.
- [9] List, P.; Joch, A.; Lainema, J.; Bjøntegaard, G.; Karczewicz, M.: Adaptive deblocking filter. *IEEE Trans. Circuits Syst. Video Technol.*, 13 (7) (2003), 614–619.
- [10] Norkin, A. *et al.*: HEVC deblocking filter. *IEEE Trans. Circuits Syst. Video Technol.*, 22 (12) (2012), 1746–1754.
- [11] Cham, W.K.; Clarke, R.J.: DC coefficient restoration in transform image coding. *IEE Proc.* 131 (1984), 709–713.
- [12] Tse, F.W.; Cham, W.K.: A DC coefficient restoration for image coding, in *Proceeding of Fifth International Conference on Image Processing and Its Application, 4–6 July, 1995*, Edinburgh, pp. 569–573, 1995.
- [13] Tse, F.W.; Cham, W.K.: DC coefficient restoration using MAP estimation technique, in *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2557–2560 (1998).
- [14] Pennebaker, W.B.; Mitchell, J.L.: *Still Image Data Compression*, Springer Verlag (1992).
- [15] Puri, A.; Schmidt, R.L.; Haskell, B.G.: Improvements in DCT based video codec. *Proc. SPIE* 3024, (1997), 676–688.
- [16] Uehara, T.; Safavi-Naini, R.; Ogunbona, P.: Recovering DC coefficients in block-based DCT. *IEEE Trans. Image Process.*, 15 (11) (2006), 3592–3596

- [17] Lakhani, G.: Predicting dc using ac coefficients for JPEG coding, *SPIE Opt. Eng.* 47 (2) (2008), 027004, 5 p.
- [18] Lakhani, G.: DCT coefficient prediction for JPEG image coding, *IEEE Int. Conf. Image Process.*, 4 (2007), 189–192.
- [19] Kondo, H.; Yamahara, K.; Liao, J.: Identification of DCT Signs for Sub-block Coding, in *IEEE International Symposium on Signal Processing and its Applications (ISSPA)*, Kuala Lumpur, Malaysia, August 2001, pp. 569–572.
- [20] Ponomarenko, N.; Bazhyna, A.; Egiazarian, K.: Prediction of signs of DCT coefficients in block-based lossy image compression, in *SPIE Proceedings, Image Processing: Algorithms and Systems V*, San Jose, California, USA, January – February 2007, 8 p.
- [21] Kazui, K. *et al*: Video coding technology proposal by Fujitsu, in *Contribution JCTVC-A115 to the 1st JCT-VC meeting*, Dresden, Germany, April 2010.
- [22] Henry, F.; Clare, G.: Residual sign prediction, in *Contribution JVET-D0031 to the 4th JVET meeting*, Chengdu, China, October 2016.
- [23] Henry, F.: Method of encoding a digital image, decoding method, devices, and associated computer programs. PCT/FR2015/051587, US 2017/0142444 A1, published in May 18, 2017.
- [24] Filippov, A.; Rufitskiy, V.; Potapov, V.: Scalar-quantization-based multi-layer data hiding for video coding applications, in *IEEE Visual Communications and Image Processing Conference (VCIP)*, Valletta, Malta, December 2014, 262–265.
- [25] Alshina, E.; Chen, J.: Test Model 1 (VTM 1) Algorithm and Encoder Description, in *Document JVET-J1002, 10th JVET meeting*, San Diego, USA, 10–20 April. 2018.
- [26] Boyce, J.; Sühring, K.; Li, X.; Seregin, V.: JVET common test conditions and software reference configurations, in *Document JVET-J1010, 10th JVET meeting*, San Diego, USA, 10–20 April, 2018
- [27] Filippov, A.; Karabutov, A.; Rufitskiy, V.; Chen, J.: CE7: Residual sign prediction in transform domain (Tests 7.5.1 and 7.5.2), in *Contribution JVET-K0044 to the 11th JVET meeting*, Ljubljana, Slovenia, July 2018.