SIViP (2013) 7:777–786
DOI 10.1007/s11760-011-0267-z

ORIGINAL PAPER

# Enhanced SATD-based cost function for mode selection of H.264/AVC intra coding

**Mohammed Golam Sarwer · Q. M. Jonathan Wu · Xiao-Ping Zhang**

Received: 10 November 2010 / Revised: 23 September 2011 / Accepted: 24 September 2011 / Published online: 7 October 2011


**Abstract** In H.264/AVC, the concept of rate-distortion optimization (RDO) mode decision has proven to be an important coding tool. However, this mode decision method makes the encoding process extremely complex, especially in the computation of the rate-distortion cost function, which includes the computations of the sum of squared difference (SSD) between the original and reconstructed image blocks and context-based entropy coding of the block. In this paper, we propose an enhanced low-complexity cost function for H.264/AVC intra $4 \times 4$ and $8 \times 8$ mode selections. The enhanced cost function uses sum of absolute Hadamard-transformed differences (SATD) and mean absolute deviation of the residual block to estimate distortion part of the cost function. A threshold-based large coefficients count is also used for estimating the bit-rate part. The proposed method improves the rate-distortion performance of conventional fast cost functions while maintaining low complexity requirement. As a result, the encoding process can be significantly accelerated with the use of the proposed cost function. Simulation results confirm that the proposed method reduces about 83.74 % computation of intra encoding of JM 12.4 encoder with negligible rate-distortion performance degradation.

**Keywords** H.264/AVC · Intra prediction · Rate-distortion optimization · Sum of absolute Hadamard-transformed differences (SATD)

## 1 Introduction

H.264/AVC [1] video coding standard is the latest international standard developed by ITU-T Video Coding Experts Group and the ISO/IEC Moving Picture Experts Group, which provides gains in compression efficiency up to 50% over a wide range of bit rates and video resolutions compared to previous standards [2]. New and advanced techniques are introduced in this new standard, such as intra prediction for I-frame encoding, multi-frames inter prediction, small block-size transform coding, context-adaptive binary arithmetic entropy coding, de-blocking filtering, etc. The rate-distortion optimization (RDO) is one of the essential parts of the H.264/AVC encoder to achieve better coding performance. However, the computational complexity of the RDO technique is extremely high, and the cost function computation makes H.264/AVC difficult to be realized in real time applications without high computing hardware.

In order to reduce the computation of rate-distortion (RD) optimization process, several rate-distortion models are proposed in literature [3–11]. To reduce the computation of intra encoding, an improved cost function for intra $4 \times 4$ mode decisions is proposed in [5]. In this cost function, sum of absolute integer transform differences (SAITD) is used in distortion part and a rate prediction algorithm is used in rate part. The major drawback of this cost function is that it requires performing the true integer transform. Even though fast transformation algorithm was proposed to perform SAITD, the overall complexity is still quite high. To reduce the complexity of rate-distortion cost computation, a fast bit-rate

M. G. Sarwer (✉) · Q. M. J. Wu
Department of Electrical and Computer Engineering,
University of Windsor, 401 Sunset Avenue, Windsor, ON, Canada
e-mail: sarwer@uwindsor.ca

Q. M. J. Wu
e-mail: jwu@uwindsor.ca

M. G. Sarwer · X.-P. Zhang
Department of Electrical and Computer Engineering,
Ryerson University, Toronto, ON, Canada

X.-P. Zhang
e-mail: xzhang@ee.ryerson.ca

estimation technique is presented to avoid the entropy coding method during intra and inter mode decision of H.264/AVC [6,9]. But this cost function still needs to calculate highly complex sum of squared differences (SSD). A bit-rate estimation method based on standard deviation of transform coefficient is proposed in [8].

In this paper, an enhanced sum of absolute Hadamard-transformed differences (SATD)-based cost function is proposed. It is well known that a block with high detail produces more distortion than a block with low details. Based on this assumption, in addition to SATD, the distortion part of the proposed cost function uses the mean absolute deviation of the residual block. If mean absolute deviation is higher, the distortion is also higher. Bit rate is predicted based on the number of large Hadamard-transformed coefficients. A block with large number of quantized Hadamard-transformed coefficients produces a high bit rate. A threshold-based large coefficient count is used for estimating the bit-rate part. The complexity of the proposed cost function is further reduced by considering only low-frequency Hadamard-transformed coefficients for distortion and bit-rate estimation. The proposed technique reduces encoding time of JM 12.4 [15] intra coding by 83.74% with acceptable performance degradation.

The remainder of this paper is organized as follows. Section 2 provides the review of rate-distortion optimized cost functions of JM 12.4 intra $4 \times 4$ mode decision techniques. In Sect. 3, SSD between original and reconstructed image block is analyzed. Section 4 introduces the proposed enhanced SATD-based cost function. The simulation results of the proposed method are presented in Sect. 5. Finally, Sect. 6 concludes the paper.

## 2 Cost functions of JM intra prediction

JM 12.4 intra prediction uses 9 prediction modes for a $4 \times 4$ luma block. To take the full advantage of all modes, the JM 12.4 encoder determines the mode that meets the best rate-distortion (RD) tradeoff using rate-distortion optimization (RDO) mode decision scheme. The best mode is the one having minimum RD cost, and this cost is expressed as

$$J_{\mathrm{RD}} = \mathrm{SSD} + \lambda \cdot R \tag{1}$$

where the SSD is the sum of squared differences between the original blocks $\boldsymbol{S}$ and the reconstructed block $\boldsymbol{C}$, and it is expressed by

$$\mathrm{SSD} = \sum_{i=1}^{4}\sum_{j=1}^{4}(S_{ij} - C_{ij})^2 \tag{2}$$

where $S_{ij}$ and $C_{ij}$ are the $(i, j)$th elements of the original block $\boldsymbol{S}$ and the reconstructed block $\boldsymbol{C}$. In (1), the $R$ is the true bits needed to encode the block and $\lambda$ is an exponential function of the quantization parameter (QP). In [12], a strong connection between the local Lagrangian multiplier and the QP was found experimentally as

$$\lambda = 0.85 \times 2^{(\mathrm{QP}-12)/3}. \tag{3}$$

Figure 1 shows the flow diagram for computations of RD cost function. It is found that the rate-distortion function $J_{\mathrm{RD}}$ introduces large computations in real encoding as it requires the following computations:

1. Compute the predicted block: $\boldsymbol{P}$
2. Compute the residual block: $\boldsymbol{E} = \boldsymbol{S} - \boldsymbol{P}$
3. Discrete cosine transform (DCT) of the residual block: $\boldsymbol{F} = \mathrm{DCT}(\boldsymbol{E})$
4. Quantize the transformed residual block: $\boldsymbol{F}'' = Q(\boldsymbol{F})$
5. Entropy coding of the quantized and transformed residual block to determine the bit-rate for encoding the block: $R = \mathrm{EC}(\boldsymbol{F}'')$
6. Inverse quantize the quantized and transformed residual block: $\boldsymbol{F}' = Q^{-1}(\boldsymbol{F}'')$
7. Inverse discrete cosine transform (IDCT) of the de-quantized block: $\boldsymbol{E}' = \mathrm{DCT}^{-1}(\boldsymbol{F}')$
8. Compute the reconstructed image block: $\boldsymbol{C} = \boldsymbol{E}' + \boldsymbol{P}$
9. Compute SSD between $\boldsymbol{S}$ and $\boldsymbol{C}$ by using (2)
10. Calculate the cost function: $J_{\mathrm{RD}} = \mathrm{SSD} + \lambda \cdot R$

The JM 12.4 encoder computes this rate-distortion optimization process for every macroblock with all possible modes. All of these processings explain the high computational complexity of $J_{\mathrm{RD}}$ cost calculation. To accelerate the coding process, JM 12.4 provides a fast SAD-based cost function:

$$J_{\mathrm{SAD}} = \mathrm{SAD} + \lambda_1 \cdot 4P \tag{4}$$

where SAD is sum of absolute differences between the original block $\boldsymbol{S}$ and the predicted block $\boldsymbol{P}$. The $\lambda_1$ is also approximate exponential function of the QP that is almost the square root of $\lambda$, and the $P = 0$ for the most probable mode and 1 for the other modes. The SAD is expressed by

$$\mathrm{SAD} = \sum_{i=1}^{4}\sum_{j=1}^{4}\left|S_{ij} - P_{ij}\right| \tag{5}$$

where $S_{ij}$ and $P_{ij}$ are the $(i, j)$th elements of the original block $\boldsymbol{S}$ and the predicted block $\boldsymbol{P}$, respectively. This SAD-based cost function saves a lot of computations as the distortion part is based on the differences between the original block and the predicted block instead of the reconstructed block. Thus, the processes of image block transformation, quantization, inverse quantization, inverse transformation and reconstruction of image block can all be saved. In addition, the rate part is pre-defined by $4P$, where

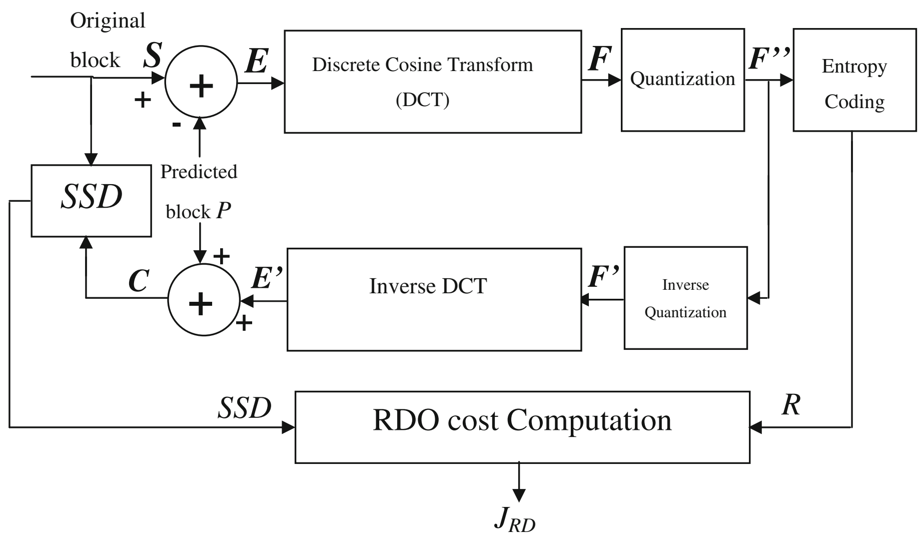



**Fig. 1** Computation of rate-distortion (RD) cost function

$P$ is equal to 0 for the most probable mode and 1 for the other modes. Thus, the context-adaptive variable-length coding (CAVLC) or context-adaptive binary arithmetic coding (CABAC) can also be saved. However, the reduction of the computations usually comes with quite significant degradation of coding efficiency.

To achieve better rate-distortion performance, JM 12.4 also provided an alternative SATD-based cost function:

$$J_{\mathrm{SATD}} = \mathrm{SATD} + \lambda_1 \cdot 4P \tag{6}$$

where SATD is sum of absolute Hadamard-transformed differences between the original block $\boldsymbol{S}$ and the predicted block $\boldsymbol{P}$, which is given by

$$\mathrm{SATD} = \sum_{i=1}^{4}\sum_{j=1}^{4} \left|h_{ij}\right| \tag{7}$$

where $h_{ij}$ are the $(i, j)$th element of the Hadamard-transformed image block $\boldsymbol{H}$. The Hadamard-transformed block $\boldsymbol{H}$ is defined as

$$\boldsymbol{H} = \boldsymbol{T}_H(\boldsymbol{S} - \boldsymbol{P})\boldsymbol{T}_H^T = \boldsymbol{T}_H(\boldsymbol{E})\boldsymbol{T}_H^T \tag{8}$$

with

$$\boldsymbol{T}_H = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \tag{9}$$

Experimental results show that the $J_{\mathrm{SATD}}$ achieve better rate-distortion performance than the $J_{\mathrm{SAD}}$, but it requires more computations due to the Hadamard transformation. If the fast Hadamard transform (FHT) is used, the computational requirement of $J_{\mathrm{SATD}}$ is reduced by 50%. However, the overall rate-distortion performance degradation is still high.

## 3 Analysis of sum of squared differences (SSD)

Before we propose the new cost function, we first present the major reason of the SSD between the original block and the reconstruction block in the rate-distortion cost function. Mathematically, the original block $\boldsymbol{S}$ and reconstructed block $\boldsymbol{C}$ can be expressed as

$$\boldsymbol{S} = \boldsymbol{E} + \boldsymbol{P} \tag{10}$$

$$\boldsymbol{C} = \boldsymbol{E}' + \boldsymbol{P} \tag{11}$$

where $\boldsymbol{P}'$ is the predicted block, $\boldsymbol{E}$ is the residual block, and $\boldsymbol{E}$ is the reconstructed residual block. Then, the SSD can be expressed as [13]

$$\begin{aligned} \mathrm{SSD} &= \|\boldsymbol{S} - \boldsymbol{C}\|_F^2 = \left\|\boldsymbol{E} + \boldsymbol{P} - \boldsymbol{E}' - \boldsymbol{P}\right\|_F^2 \\ &= \left\|\boldsymbol{E} - \boldsymbol{E}'\right\|_F^2 = \left\|\boldsymbol{F} - \boldsymbol{F}'\right\|_F^2 \end{aligned} \tag{12}$$

where $\|\|_F$ is the Frobenius norm, and $\boldsymbol{F}$ and $\boldsymbol{F}'$ are the DCT-transformed residual block and inverse quantized transformed residual block. Here, SSD calculated from $\boldsymbol{S}$ and $\boldsymbol{C}$ is called spatial-domain SSD and SSD computed from $\boldsymbol{F}$ and $\boldsymbol{F}'$ is called DCT-domain SSD since SSD is performed after DCT transform (clearly shown in Fig. 1). From (12), it is clear that the spatial-domain SSD is equivalent to DCT-domain SSD. Based on this relationship, we can calculate the R-D cost in DCT transform domain with $J_{\mathrm{RD}} = \mathrm{SSD}(\boldsymbol{F}, \boldsymbol{F}') + \lambda \cdot R$. From Fig. 1, it is evident that DCT-domain SSD can avoid inverse DCT transform of image block during RD cost computation. However, this computational reduction is very limited.

Equation 12 shows that SSD is due to the difference between $\boldsymbol{F}$ and $\boldsymbol{F}'$. This difference happens due to quantization. Thus, the reason of the SSD is due to the quantization errors in the DCT-transformed residual block $\boldsymbol{F}$. SATD uses the sum of the absolute coefficient values of $\boldsymbol{H}$ to estimate the distortion part, where SAD uses the sum of the absolute coefficients of residual error $\boldsymbol{E}$. As compared to residual error $\boldsymbol{E}$, $\boldsymbol{H}$ is quite closer to $\boldsymbol{F}$ because property of the Hadamard transform is quite close to the DCT transform used in H.264/AVC. Therefore, SATD can perform better than SAD for the distortion part estimation of the rate-distortion cost function. In this paper, we use Hadamard-transformed coefficients to define an enhanced cost function, which maintains similar complexity of the SATD but with better rate-distortion performance.

## 4 Enhanced SATD-based cost function

It is reasonable to say that a complex block produces a large distortion value compared to a simple block. In other words, residual block with high details has larger distortion value than homogeneous block. Let us consider two residual blocks $\boldsymbol{E}_A$ and $\boldsymbol{E}_B$.

$$\boldsymbol{E}_A = \begin{bmatrix} 0 & 10 & 8 & 10 \\ 9 & 7 & 4 & 10 \\ 1 & 10 & 11 & 4 \\ 19 & 6 & 15 & 7 \end{bmatrix} \text{ and } \boldsymbol{E}_B = \begin{bmatrix} 22 & 22 & 22 & 22 \\ 22 & 22 & 22 & 22 \\ 20 & 20 & 20 & 20 \\ 22 & 22 & 22 & 22 \end{bmatrix}$$

Assume QP $= 24$. If we compute SSD based on Fig. 1, it is found that $\text{SSD}_A = 173$ and $\text{SSD}_B = 48$. That means distortion of block $A$ is higher than that of block $B$. This is understandable because block $A$ contains larger detail than block $B$. But if we calculate SATD of $A$ and $B$ by (7), we found that both of the residual blocks produce same SATD value $\text{SATD}_A = \text{SATD}_B = 368$. This means only SATD is not enough to measure the distortion.

Figure 2 shows the probability distribution of the residual coefficients with different value of rate-distortion cost function. Data are collected by encoding 20 frames of three different types of video sequences (*Akiyo, Foreman* and *Stefan*) with different QPs. They correspondingly represent different kinds of video: slow motion, medium motion, and fast motion. For example "*Akiyo*" is a sequence of low spatial details that contains low changes in motion. "*Foreman*" is a sequence with medium changes in motion and contains dominant luminance changes. "*Stefan*" contains panning motion and high spatial color details. Four distributions are plotted, and each has different RD cost value. For example, red line of Fig. 2 indicates the probability distribution function of those residual coefficients that produce $J_{\text{RD}}$ lower than 300. Similarly, the probability distribution of the coefficients with higher RD cost values ($J_{\text{RD}} > 800$) is plotted by black line. As shown in Fig. 2, the probability distribution function (PDF) of the residual coefficients is becoming wider with an increment of RD cost function. That means the variance of residual coefficients is increased with increment of RD cost value.

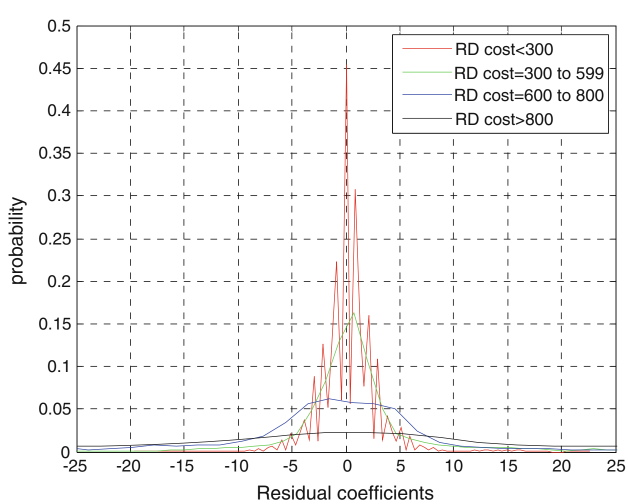



**Fig. 2** Probability distribution of the residual coefficients $E_{ij}$

Therefore, the distortion part of the proposed cost function is defined as follows

$$\text{ESATD} = \text{SATD} + \alpha \times \sigma(\boldsymbol{E}) \tag{13}$$

where ESATD is enhanced SATD, $\alpha$ is a constant value, and $\sigma(\boldsymbol{E})$ is the mean absolute deviation of the residual block $\boldsymbol{E}$, which represent the variation of pixel values. $\sigma(\boldsymbol{E})$ of a $4 \times 4$ block is defined as

$$\begin{aligned} \sigma(\boldsymbol{E}) &= \frac{1}{16} \sum_{i=1}^{4} \sum_{j=1}^{4} \left| E_{ij} - \mu \right| \\ \text{with } \mu &= \frac{1}{16} \sum_{i=1}^{4} \sum_{j=1}^{4} E_{ij} = \frac{h_{11}}{16} = (h_{11} \gg 4) \end{aligned} \tag{14}$$

Where $h_{11}$ is the (1,1)th coefficient of Hadamard-transformed matrix $\boldsymbol{H}$. No extra computation is required to compute $h_{11}$ because it is already computed for SATD calculation. The proposed cost function also uses a rate predictor for estimating the rate for encoding the residual block instead of just using a penalty cost ($4P$). The rate predictor is based on the property of the context-based adaptive variable-length coding (CAVLC) entropy coder. To avoid DCT, quantization, and CAVLC encoding processes during RDO, the proposed rate predictor only uses the total number of the non-zero quantized Hadamard-transformed coefficients ($T_{\text{bc}}$) that can be obtained by performing simple threshold and counter operations. Based on the property of the CAVLC's VLC tables [1], larger $T_{\text{bc}}$ will produce more encoded bits. With the penalty cost ($4P$) from $J_{\text{SAD}}$, the overall estimated rate $R_e$ is defined as

$$R_e = \beta T_{\text{bc}} + 4P \tag{15}$$

Where $\beta$ is the proportionality constant. In order to set the weighting factors $\beta$, we have done several experiments for

**Table 1** Quantization step sizes $Q_{step}$ in H.264/AVC codec

| QP | $Q_{step}$ |
|---|---|
| 0 | 0.625 |
| 1 | 0.6875 |
| 2 | 0.8125 |
| 3 | 0.875 |
| 4 | 1 |
| 5 | 1.125 |
| 6 | 1.25 |

different video sequences (*Akiyo, Foreman,* and *Stefan*) with QCIF format at different QP values. By varying the value of $\beta$ from 1 to 10, we observed the rate-distortion performance of these video sequences. We found that $\beta = 3$ produced a better rate-distortion performance.

The total number of the non-zero quantized Hadamard-transformed coefficients of a 4 × 4 block ($T_{bc}$) is calculated as follows:

Step1: $T_{bc} = 0$;
Step 2: For $i = 1$ to 4
 For $j = 1$ to 4
  If ($|h_{ij}| \geq Q_{step}$) then $T_{bc} = T_{bc} + 1$
 End for
End for

$Q_{step}$ is the quantization step size used in H.264/AVC encoder. Values of $Q_{step}$ with seven different QPs are given in Table 1. $Q_{step}$ doubles in size for every increment of 6 in QP [14]. For simplicity, we did not consider the effect of dead zone while counting the number of non-zero Hadamard-transformed coefficients. The proposed cost function is defined as

$$J_{ESATD} = \text{ESATD} + \lambda_1 R_e \tag{16}$$

By substituting the value of ESATD and $R_e$, the cost function becomes

$$J_{ESATD} = \text{SATD} + \alpha \times \sigma(E) + \lambda_1(3T_{bc} + 4P) \tag{17}$$

In order to find the value of $\alpha$, we have done some simulations with different types of sequences by varying QPs and better results were found at $\alpha = 1.25$. As explained in the previous section, the reason of the SSD is due to the quantization error of the DCT-transformed coefficients of the residual block $E$, which can be estimated by sum of absolute coefficients of $H$ in SATD. It is clear that high frequency coefficients of Hadamard-transformed coefficients are insignificant and bear a low value.

Figure 3 shows the zig-zag scan and the corresponding value of frequency of the Hadamard-transformed matrix $\boldsymbol{H}$ of a 4 × 4 block. $\boldsymbol{H}$ can be redefined in terms of frequency as follows.

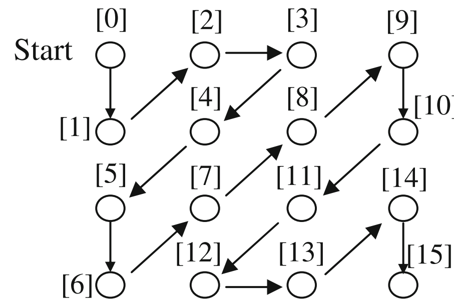



**Fig. 3** Zig-zag scan and corresponding frequency of $\boldsymbol{H}$

$$\boldsymbol{H} = \begin{bmatrix} h_{11} & h_{12} & h_{13} & h_{14} \\ h_{21} & h_{22} & h_{23} & h_{24} \\ h_{31} & h_{32} & h_{33} & h_{34} \\ h_{41} & h_{42} & h_{43} & h_{44} \end{bmatrix} = \begin{bmatrix} I_0 & I_2 & I_3 & I_9 \\ I_1 & I_4 & I_8 & I_{10} \\ I_5 & I_7 & I_{11} & I_{14} \\ I_6 & I_{12} & I_{13} & I_{15} \end{bmatrix} \tag{18}$$

where $I_f$ is the Hadamard-transformed coefficient of frequency $f$. After Hadamard transform, the high frequency coefficients usually have small energy. Figure 4 shows the average intensity distribution of Hadamard-transformed coefficients of 4 × 4 blocks of two different types of video sequences. It is clear that the intensity values of high frequency coefficients are negligible. Based on this observation, the proposed cost function calculates only 10 low-frequency Hadamard coefficients instead of calculating all 16 coefficients.

Therefore, the proposed cost function becomes,

$$J_{ESATD} = \text{SATD}' + \alpha \times \sigma(E) + \lambda_1(3T'_{bc} + 4P) \tag{19}$$

$$\text{with, SATD}' = \sum_{f=0}^{9} |I_f|$$

The step by step computation of the proposed cost function is provided below:

1. Compute the predicted block: $\boldsymbol{P}$
2. Compute the residual block: $\boldsymbol{E} = \boldsymbol{S} - \boldsymbol{P}$
3. Compute only 10 low-frequency coefficients of Hadamard-transformed matrix of the residual block: $\boldsymbol{H} = HT(\boldsymbol{E})$
4. Calculate the SATD′ and $T'_{bc}$
 Set SATD′ = 0 and $T'_{bc} = 0$
  For $f = 0\ to\ 9$
   $\text{SATD}' = \text{SATD}' + |I_f|$;
   If ($|I_f| \geq Q_{step}$) $T'_{bc} = T'_{bc} + 1$;
  End for
5. Calculate $\sigma(E)$ by (14), where $\mu = (h_{11} \gg 4) = (I_0 \gg 4)$
6. Calculate $J_{ESATD} = \text{SATD}' + \alpha \times \sigma(\boldsymbol{E}) + \lambda_1(3T'_{bc} + 4P)$, where $\alpha = 1.25$ and $P$ equal to 0 for the most probable mode and 1 for the other modes.

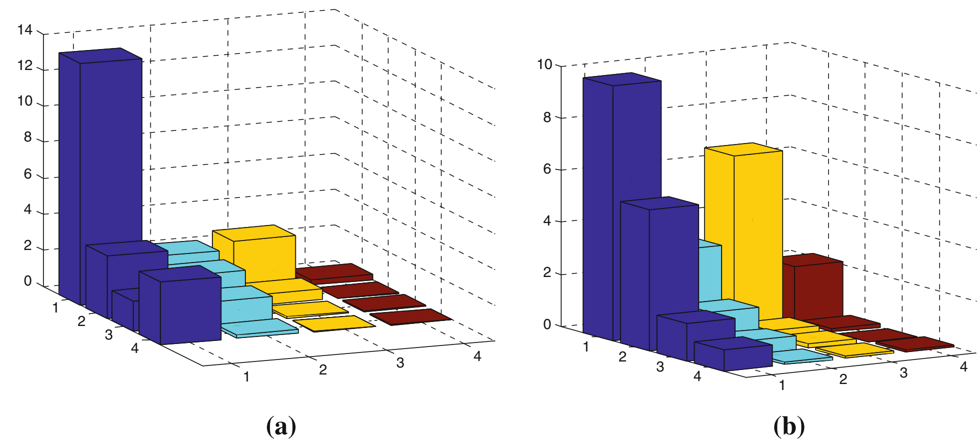


**Fig. 4** Average intensity distribution of the Hadamard transform coefficients $\boldsymbol{H}$. **a** Akiyo, **b** Stefan

If we use fast Hadamard transform, 16 coefficients of $\boldsymbol{H}$ can be calculated by 64 additions. Therefore, in order to calculate SATD, total 96 operations (64 additions for 16 coefficients + 16 additions for sum + 16 absolute operations) are required. But computing SATD′ needs total 74 operations (54 additions for 10 coefficients + 10 additions for sum + 10 absolute operations). Thus, SATD′ can save 20 operations for each $4 \times 4$ block.

The $8 \times 8$ intra mode decision method of JM 12.4 encoder also computes 9 prediction modes. The prediction directions are exactly same as that of $4 \times 4$ mode. The proposed enhanced SATD-based cost function can also be applied to $8 \times 8$ intra mode selection of JM 12.4 encoder. For $8 \times 8$ modes, $8 \times 8$ Hadamard transform is used for SATD computation. Similar to (18), Hadamard-transformed matrix of $8 \times 8$ modes, $\boldsymbol{H}_{8\times8}$, can be defined in terms of frequency as follows

$$\boldsymbol{H}_{8\times8} = \begin{bmatrix} I_0 & I_2 & I_3 & I_9 & I_{10} & I_{20} & I_{21} & I_{35} \\ I_1 & I_4 & I_8 & I_{11} & I_{19} & I_{22} & I_{34} & I_{36} \\ I_5 & I_7 & I_{12} & I_{18} & I_{23} & I_{33} & I_{37} & I_{48} \\ I_6 & I_{13} & I_{17} & I_{24} & I_{32} & I_{38} & I_{47} & I_{49} \\ I_{14} & I_{16} & I_{25} & I_{31} & I_{39} & I_{46} & I_{50} & I_{57} \\ I_{15} & I_{26} & I_{30} & I_{40} & I_{45} & I_{51} & I_{56} & I_{58} \\ I_{27} & I_{29} & I_{41} & I_{44} & I_{52} & I_{55} & I_{59} & I_{62} \\ I_{28} & I_{42} & I_{43} & I_{53} & I_{54} & I_{60} & I_{61} & I_{63} \end{bmatrix} \tag{20}$$

Similar to $4 \times 4$ mode decision, instead of using 64 coefficients of $8 \times 8$ block, only low-frequency Hadamard-transformed coefficients ($f = 0$ to 35) are used for computation of the proposed cost function. For $8 \times 8$ intra modes, the proposed cost function is defined as follows.

$$J_{8\times8(\mathrm{ESATD})} = \mathrm{SATD}'_{8\times8} + 4\alpha \times \sigma_{8\times8}(\boldsymbol{E}_{8\times8}) + \lambda_1(3T'_{8\times8(\mathrm{bc})} + 4P) \tag{21}$$

with $\mathrm{SATD}'_{8\times8} = \sum_{f=0}^{35} |I_f|$ and $\sigma_{8\times8}(\boldsymbol{E}_{8\times8})$

$$= \frac{1}{64} \sum_{i=1}^{8} \sum_{j=1}^{8} |E_{ij} - \mu_{8\times8}| \tag{22}$$

$$\mu_{8\times8} = \frac{1}{64} \sum_{i=1}^{8} \sum_{j=1}^{8} E_{ij} = \frac{h_{8\times8_{11}}}{64} = (h_{8\times8_{11}} \gg 6) \tag{23}$$

$T'_{8\times8(\mathrm{bc})}$ is the total number of low-frequency ($f = 0\, to\, 35$) non-zero coefficients of a $8 \times 8$ block, $\alpha = 1.25$ and $P$ equal to 0 for the most probable mode and 1 for the other modes.

## 5 Simulation results

The performance of the proposed cost function was tested using the first 100 frames of different types of video sequences (*Akiyo, Claire, News, Container, Foreman, Stefan, Mother_daughter, Silent, Hall, Flower*, and *Ice*). Within these sequences, *Akiyo, Foreman*, and *Stefan* sequences were used for training for parameter selection. They correspondingly represent different kinds of video: slow motion, medium motion, and fast motion. The experiment was carried out in the JVT JM 12.4 [15] encoder, and the test parameters are listed as below:

- CAVLC is enabled;
- Frame rate is 30;
- All frames are I frames;
- QPs are 30/36/42/48
- Number of frames: 100

The RD performances of the proposed method are measured in terms of PSNR and bit-rate differences with JM 12.4 encoder. PSNR and bit-rate differences are calculated according to the numerical averages between RD curves derived

**Table 2** PSNR and bit rate comparison with JM 12.4 (only 4 × 4 modes are enabled)

| Sequence | ΔPSNR in dB | | | Δ*R*% | | |
|---|---|---|---|---|---|---|
| | $J_{SAD}$ | $J_{SATD}$ | $J_{ESATD}$ | $J_{SAD}$ | $J_{SATD}$ | $J_{ESATD}$ |
| Akiyo QCIF | −0.39 | −0.37 | −0.12 | 7.49 | 6.86 | 2.71 |
| Foreman QCIF | −0.37 | −0.21 | −0.07 | 8.46 | 4.76 | 2.34 |
| Container QCIF | −0.36 | −0.30 | −0.18 | 8.83 | 7.26 | 4.29 |
| Claire QCIF | −0.33 | −0.31 | −0.01 | 5.96 | 5.64 | 1.52 |
| Stefan QCIF | −0.60 | −0.47 | −0.27 | 16.84 | 12.99 | 7.43 |
| News QCIF | −0.49 | −0.42 | −0.19 | 10.79 | 9.11 | 4.82 |
| Mother_daughter QCIF | −0.29 | −0.22 | −0.07 | 6.90 | 4.86 | 2.13 |
| Silent CIF | −0.28 | −0.21 | −0.12 | 8.90 | 6.20 | 4.82 |
| Hall CIF | −0.36 | −0.35 | −0.14 | 6.41 | 6.29 | 2.59 |
| Flower CIF | −0.37 | −0.26 | −0.17 | 7.69 | 5.90 | 3.99 |
| Ice 704 × 576 | −0.35 | −0.30 | −0.18 | 11.89 | 6.12 | 3.43 |
| Average | −0.38 | −0.31 | −0.14 | 9.10 | 6.90 | 3.64 |

from the JM 12.4 and the proposed algorithm, respectively. The detail procedure to calculate these differences can be found in [16]. In order to evaluate complexity reduction of the proposed method as compared to the JM 12.4 encoder, $\Delta T_1$ (%) is defined as follows

$$\Delta T_1 = \frac{T_{JM} - T_{proposed}}{T_{JM}} \times 100\% \quad (24)$$

where, $T_{JM}$ and $T_{proposed}$ denote the total encoding time of the JM 12.4 encoder with $J_{RD}$ and with the proposed cost function, respectively.

### 5.1 Experiments with only 4 × 4 intra modes

In these experiments, four cost functions ($J_{RD}$, $J_{SAD}$, $J_{SATD}$, and $J_{ESATD}$) are simulated and only 4 × 4 intra modes are enabled. The PSNR and bit-rate comparisons of the proposed cost function are tabulated in Table 2. The positive values mean increments whereas negative values mean decrements. It is shown that RD performance degradation of both $J_{SAD}$ and $J_{SATD}$ is significant. In case of $J_{SATD}$, the average PSNR reduction is about 0.31 dB and average bit-rate increment is about 6.90%. Whereas in our proposed method, the average PSNR reduction is about 0.14 dB and average bit-rate increment is about 3.64%.

The coding results of the proposed method are very similar to actual RDO method of JM 12.4. It is clear that our proposed cost function always performs better than $J_{SATD}$ and $J_{SAD}$. The worst case is encoding of *Stefan* video sequence. For *Stefan*, the proposed method increases the bit rate of about 7.43% whereas $J_{SATD}$ generates around 12.99% of bit-rate increment. The RD curves of four different types of video sequences are presented Fig. 5. It is shown that RD curve of the proposed method is much closer to RD optimized curve and better than $J_{SATD}$ and $J_{SAD}$ for all types of sequences.

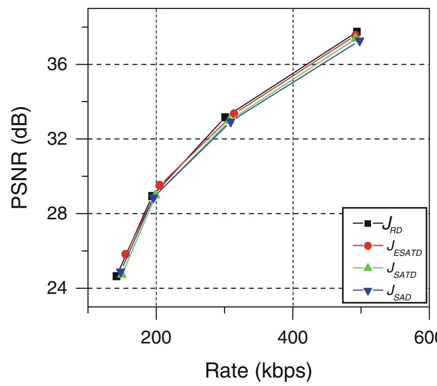
The complexity reductions of $J_{SAD}$, $J_{SATD}$, and $J_{ESATD}$ are tabulated in Table 3. The proposed cost function does not compute DCT, quantization, entropy coding, inverse quantization, and inverse DCT during mode selection. Therefore, lots of computations can be saved. The proposed algorithm reduced about 84.67% of total encoding time compared to rate-distortion optimized cost function $J_{RD}$. The computational reduction of the proposed method is almost similar to $J_{SATD}$. However, the rate-distortion performance of the proposed method is much better as compared with $J_{SATD}$.

### 5.2 Experiments with all intra modes

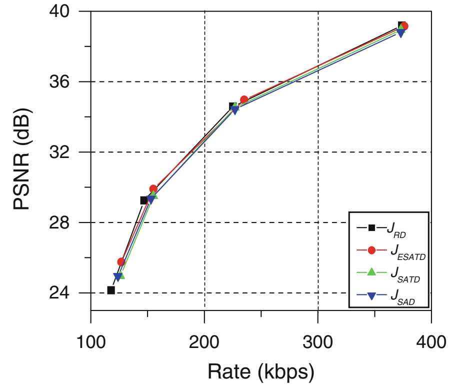
In this experiment, all 100 frames are encoded by intra coding and all intra modes (4 × 4, 8 × 8, and 16 × 16) are enabled. The proposed cost function is applied for 4 × 4 and 8 × 8 modes. Since computational complexity of 16 × 16 modes are not high because only 4 modes are used, the proposed cost function was not integrated with 16 × 16 mode. The experimental results are tabulated in Tables 4 and 5. We can see that the average PSNR degradation is in the range of 0.16 dB and bit-rate degradation is 4.17 %, with a maximum for sequence *Stefan* with 0.35 dB and 8.26%. In case of $J_{SATD}$, average PSNR and bit-rate degradation is about 0.29 dB and 6.33%, respectively. We can see that the proposed method reduces 83.74% computations of original JM 12.4 encoder.

### 5.3 Comparison with other method

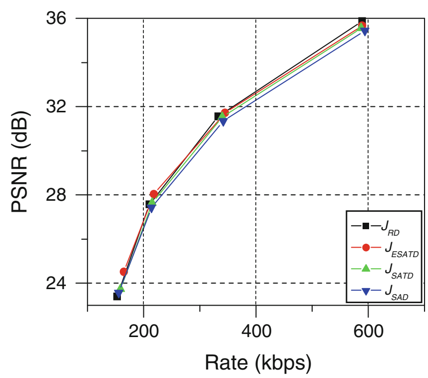
In this experiment, the proposed method is compared with fast SAITD-based cost function ($J_{SAITD}$) proposed in [5] in terms of rate-distortion performance and complexity. Only 4 × 4 intra modes are enabled in this simulation. Table 6



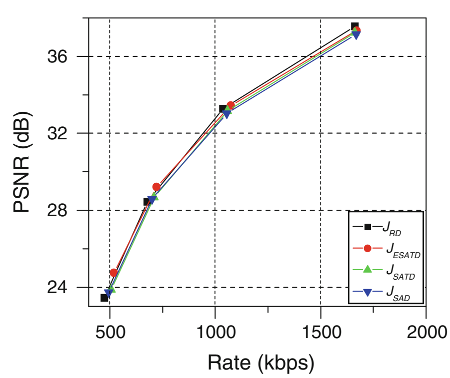
**(a)** Akiyo QCIF



**(b)** Claire QCIF



**(c)** Foreman QCIF



**(d)** Hall CIF

**Fig. 5** Rate-distortion (RD) curves of four different cost functions (only 4 × 4 modes are enabled)

**Table 3** Complexity comparison with JM 12.4 (only 4 × 4 modes are enabled)

| Sequence | $\Delta T_1\%$ | | |
|---|---|---|---|
| | $J_{SAD}$ | $J_{SATD}$ | $J_{ESATD}$ |
| Akiyo (QCIF) | 86.76 | 85.92 | 85.76 |
| Foreman (QCIF) | 87.33 | 86.40 | 86.17 |
| Container QCIF | 85.49 | 84.65 | 83.99 |
| Claire QCIF | 84.03 | 83.15 | 82.49 |
| Stefan QCIF | 88.88 | 88.17 | 87.36 |
| News QCIF | 85.66 | 84.99 | 84.30 |
| Mother_daughter QCIF | 84.17 | 83.38 | 82.57 |
| Silent CIF | 87.43 | 86.99 | 86.67 |
| Hall CIF | 90.80 | 86.39 | 85.78 |
| Flower CIF | 86.36 | 85.18 | 83.54 |
| Ice 704 × 576 | 84.74 | 83.33 | 82.69 |
| Average | 86.51 | 85.32 | 84.67 |

shows the comparison results. In Table 6, PSNR and bit-rate performances are calculated based on [16] and complexity reduction is calculated as follows:

$$\text{Complexity reduction } (\Delta T_2\%) = \frac{T_{\text{ref}[5]} - T_{\text{proposed}}}{T_{\text{ref}[5]}} \times 100\% \quad (25)$$

Where $T_{\text{ref}[5]}$ and $T_{\text{proposed}}$ are the total encoding time of the method presented in [5] and the proposed method, respectively. From the comparison results, it is shown that our proposed method reduces the bit rate by 1.58% and increases the PSNR by 0.07 dB on average. The proposed cost function not only improves the RD performance but is also 15.88% faster than $J_{\text{SAITD}}$.

## 6 Conclusion

In this paper, a simple and fast cost function based on SATD is proposed for intra mode decision of H.264/AVC. The distortion part of the cost function is estimated based on the SATD and mean absolute deviation of residual block. If mean absolute deviation is higher, the distortion is also higher. Bit rate is predicted based on the number of large Hadamard-transformed coefficients. A block with large number of quantized Hadamard-transformed coefficients produces large bit rate. The experimental results verified that the proposed technique is very suitable for intra mode decision of H.264/AVC. With the proposed scheme, DCT, quantization, inverse quantization, and inverse DCT operations can be skipped during the mode decision process. The proposed technique reduces encoding time of JM 12.4 intra encoding by 83.74% with

**Table 4** PSNR and bit rate comparison with JM 12.4 (all intra modes are enabled)

| Sequence | ΔPSNR in dB | | | $\Delta R\%$ | | |
|---|---|---|---|---|---|---|
| | $J_{\text{SAD}}$ | $J_{\text{SATD}}$ | $J_{\text{ESATD}}$ | $J_{\text{SAD}}$ | $J_{\text{SATD}}$ | $J_{\text{ESATD}}$ |
| Akiyo QCIF | −0.28 | −0.21 | −0.15 | 5.36 | 4.58 | 2.91 |
| Foreman QCIF | −0.59 | −0.29 | −0.12 | 8.32 | 6.21 | 4.98 |
| Container QCIF | −0.41 | −0.27 | −0.21 | 7.69 | 5.23 | 4.10 |
| Claire QCIF | −0.29 | −0.21 | −0.04 | 5.25 | 3.10 | 1.91 |
| Stefan QCIF | −0.73 | −0.51 | −0.35 | 15.24 | 13.20 | 8.26 |
| News QCIF | −0.47 | −0.37 | −0.23 | 12.25 | 8.87 | 3.52 |
| Mother_daughter QCIF | −0.27 | −0.22 | −0.12 | 7.96 | 5.23 | 3.58 |
| Silent CIF | −0.36 | −0.23 | −0.14 | 7.41 | 5.20 | 4.10 |
| Hall CIF | −0.30 | −0.24 | −0.16 | 6.14 | 5.41 | 4.28 |
| Flower CIF | −0.41 | −0.35 | −0.21 | 8.10 | 6.58 | 4.23 |
| Ice 704 × 576 | −0.31 | −0.27 | −0.13 | 8.69 | 6.11 | 4.01 |
| Average | −0.40 | −0.29 | −0.16 | 8.40 | 6.33 | 4.17 |

**Table 5** Complexity comparison with JM 12.4 (all intra modes are enabled)

| Sequence | $\Delta T_1\%$ | | |
|---|---|---|---|
| | $J_{\text{SAD}}$ | $J_{\text{SATD}}$ | $J_{\text{ESATD}}$ |
| Akiyo (QCIF) | 89.12 | 86.24 | 85.01 |
| Foreman (QCIF) | 87.32 | 85.02 | 83.58 |
| Container QCIF | 87.14 | 85.93 | 82.19 |
| Claire QCIF | 86.87 | 84.22 | 83.19 |
| Stefan QCIF | 87.12 | 86.00 | 84.53 |
| News QCIF | 84.92 | 83.33 | 83.10 |
| Mother_daughter QCIF | 85.27 | 83.65 | 82.11 |
| Silent CIF | 88.25 | 87.36 | 84.99 |
| Hall CIF | 87.10 | 86.28 | 85.03 |
| Flower CIF | 89.26 | 86.22 | 84.17 |
| Ice 704 × 576 | 85.23 | 85.01 | 83.27 |
| Average | 87.05 | 85.38 | 83.74 |

**Table 6** Comparison with $J_{\text{SAITD}}$ [5]

| Sequence | ΔPSNR | $\Delta R\%$ | $\Delta T_2\%$ |
|---|---|---|---|
| Akiyo (QCIF) | 0.10 | −1.93 | 16.24 |
| Foreman (QCIF) | 0.04 | −0.89 | 18.69 |
| Container QCIF | 0.05 | −1.48 | 14.23 |
| Claire QCIF | 0.11 | −1.85 | 12.77 |
| Stefan QCIF | 0.08 | −1.99 | 11.60 |
| News QCIF | 0.12 | −2.66 | 13.62 |
| Mother_daughter QCIF | 0.09 | −1.87 | 12.98 |
| Silent CIF | 0.04 | −1.22 | 21.35 |
| Hall CIF | 0.06 | −1.15 | 22.59 |
| Flower CIF | 0.05 | −1.43 | 13.79 |
| Ice 704 × 576 | 0.07 | −0.98 | 16.89 |
| Average | 0.07 | −1.58 | 15.88 |

acceptable performance degradation. The RD performance and computational complexity of this algorithm are also better than fast SAITD-based cost function.

**Acknowledgments** The work described in this paper has been supported by National Science and Engineering Research Council (NSERC), Canada.

## References

1. Draft ITU-T Recommendation and Final Draft International Standard of joint video specification (ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC), documents: JVT-G050, 7th Meeting, Pattaya, Thailand, 7–14 March, 2003
2. Ostermann, J., Bormans, J., List, P., Marpe, P., Narroschke, M., Pereira, F., Stockhammer, T., Wedi, T.: Video Coding with H.264/AVC: Tools, Performance, and Complexity. IEEE Circuits Syst. Mag. **4**, 7–28 (2004)
3. Chiang, T., Zhang, Y.Q.: A new rate control scheme using quadratic rate distortion model. IEEE Trans. Circuits Syst. Video Tech. **7**(1), 246–250 (1997)
4. Corbera, J.R., Lei, S.: Rate control of DCT video coding for low-delay communication. IEEE Trans. Circuits Syst. Video Tech. **9**(1), 172–185 (1999)
5. Tseng, C.H., Wang, H.M., Yang, J.F.: Enhanced Intra 4 × 4 Mode Decision for H.264/AVC Coders. IEEE Trans. Circuits Syst. Video Tech. **16**(8), 1027–1032 (2006)
6. Sarwer, M.G., Po, L.M.: Fast bit rate estimation for mode decision of H.264/AVC. IEEE Trans. Circuits Syst. Video Tech. **17**(10), 1402–1407 (2007)
7. Kuang, Y., Yang, J.F., Sun, M.T.: Efficient rate-distortion estimation for H.264/AVC coders. IEEE Trans. Circuits Syst. Video Tech. **16**(5), 600–611 (2006)

8. Lee, Y.M., Sun, Y.T., Lin, Y.: SATD-based intra mode decision for H.264/AVC video coding. IEEE Trans. Circuits Syst. Video Tech. **20**(3), 463–469 (2010)
9. Sarwer M.G., Po L.M.: (2007) Bit Rate estimation for cost function of 4 × 4 intra mode decision of H.264/AVC. Proc. of IEEE International Conference on Multimedia and Expo (ICME 2007), Beijing, China, July 2–5, pp. 1579–1582
10. Wang, O., Zhao, D., Gao, W., Ma, S.: Low complexity RDO mode decision based on a fast coding-bits estimation model for H.264/AVC. Proc. IEEE International Symposium on Circuits and Systems, 2005, Kobe, Japan, May 23–26, 2005, vol. 4, pp. 3467–3470.
11. Hahm, J., Kyung, C.M.: Efficient CABAC rate estimation for H.264/AVC mode decision. IEEE Trans. Circuits Syst. Video Tech. **20**(2), 310–316 (2010)
12. Sullivan, G.J., Wiegand, T.: Rate-distortion optimization for video compression. IEEE Signal Process. Mag. **15**, 74–90 (1998)
13. Po, L.M., Guo, K.: Transform-domain fast sum of the squared difference computation for H.264 rate-distortion optimization. IEEE Trans. Circuits Syst. Video Tech. **17**(6), 765–773 (2007)
14. Richardson, I.E.G.: H.264 and MPEG-4 video compression—video coding for next generation multimedia, Ch. 6. pp. 191–192. Wiley, New York (2003)
15. Joint Video Team (JVT) reference software version 12.4, available online: http://iphome.hhi.de/suehring/tml/download/old_jm/
16. Bjontegaard, G.: Calculation of average PSNR differences between RD-curves. Presented at the 13th VCEG-M33 Meeting, Austin, TX (2001)