



Available online at www.sciencedirect.com

SCIENCE @ DIRECT®

JOURNAL OF
VISUAL
Communication &
IMAGE
Representation

J. Vis. Commun. Image R. 17 (2006) 291–310

www.elsevier.com/locate/jvcir

Fast H.264 Intra-prediction mode selection using joint spatial and transform domain features

Changsung Kim ^a, Hsuan-Huei Shih ^b, C.-C. Jay Kuo ^{a,*}

^a *Integrated Media Systems Center, Department of Electrical Engineering,
University of Southern California, Los Angeles, CA 90089-2564, USA*

^b *MAVs Lab. Inc., 3F-2, No. 185, Kur-Wong Road, Aspire Park, Lung-Tan, Tau-Yuan 325, Taiwan*

Received 26 April 2004; accepted 10 May 2005

Available online 1 July 2005

Abstract

A fast H.264 Intra-prediction mode selection scheme is proposed in this work. The objective is to reduce the encoder complexity without significant rate–distortion performance degradation. The proposed method uses spatial and transform domain features of the target block jointly to filter out the majority of candidate modes. This is justified by examining the posterior error probability and the average rate–distortion loss. For the final mode selection, either the feature-based or the RDO (rate–distortion optimization)-based method is applied to 2–3 candidate modes. It is demonstrated by experimental results that the proposed scheme demands only 7–10% of the complexity of the RDO (rate–distortion optimized) mode decision scheme with little quality degradation.

© 2005 Elsevier Inc. All rights reserved.

Keywords: Video coding; H.264; AVC; Intra-mode decision; Intra prediction

* Corresponding author. Fax: +1 213 740 4651

E-mail addresses: changasuk@sipi.usc.edu (C. Kim), maverick@mavslab.com (H.-H. Shih), ckuo@sipi.usc.edu (C.-C. Jay Kuo).

1. Introduction

The emerging video coding standard H.264, which is jointly developed by ITU-T and MPEG, provides the state-of-the-art video coding technique to meet a wide range of applications [1]. H.264 offers a significant performance improvement over previous video coding standards such as H.263++ and MPEG-4 [2,3] at the expense of a higher computational complexity. Among many new features, Inter and Intra-prediction modes are greatly enriched using variable block sizes and various directional predictions, respectively [4,5]. The Intra-prediction technique is recognized to be one of the main factors that contribute to the success of H.264. It is reported in [6] that H.264 Intra-frame coding outperforms the JPEG-2000 still image compression standard due to this feature.

The RD optimization (RDO) technique [7] has been employed in H.264 for Intra-prediction mode selection to achieve coding efficiency. However, the computational complexity of the RDO technique is extremely high since the encoder has to encode the target block by searching all possible modes exhaustively for the best mode in the RD sense [8]. Efforts have been made in developing fast H.264 Intra-mode prediction algorithms. Pan et al. [8] proposed a fast mode decision scheme with a pre-processing technique, which measures the edge direction of a given block so as to reduce the number of probable modes for complexity reduction. The performance of this method is about 20–30% (or 55–65%) faster than the RDO method at the cost of 2% (or 5%) extra bits. Jeon and Lee [9] proposed another fast Intra-mode decision scheme, where the encoding speed is approximately 30% faster than that of the RDO method.

In this work, we present a simple yet effective fast mode decision algorithm for H.264 Intra prediction using spatial and transform domain features of the target block jointly [10]. This method is able to filter out the majority of candidate modes so that we only have to focus 2–3 modes for the final mode selection. To justify the use of joint features, two quantities are measured and analyzed. They are the posterior error probability and the average rate–distortion loss. Furthermore, to speed up the RDO process in the ultimate mode selection, an RD model is developed [11]. The proposed mode decision scheme has been integrated with the H.264 JM7.3a codec for performance evaluation. It is compared with the RDO mode decision scheme of H.264 in terms of the computational complexity, the average PSNR and the coding bit rates for several test sequences [12]. Simulation results demonstrate that the RD performance of the proposed algorithms is almost identical with that of the RDO mode decision scheme for a wide range of bit rates. However, the proposed algorithm only demands about 7–10% of the complexity of the H.264 RDO method.

The rest of the paper is organized as follows. The Intra prediction of H.264 is reviewed in Section 2. Two features for mode filtering are introduced in Section 3. The mode filtering using joint features is described in Section 4. The final mode selection is discussed in Section 5. Experimental results are provided in Section 6 to show the performance of the proposed scheme. Concluding remarks are given in Section 7.

2. Intra prediction for H.264

The H.264 standard exploits the spatial correlation between adjacent macroblocks/blocks for Intra prediction. That is, the current macroblock/blocks is predicted by adjacent pixels in the upper and the left macroblocks/blocks that are decoded already. H.264 offers a rich set of prediction patterns for Intra prediction. They are nine prediction modes for 4×4 luma blocks and four prediction modes for 16×16 luma blocks. Each mode follows a certain prediction direction and the predicted samples are obtained by a weighted average of decoded values of neighboring macroblocks/blocks [7].

To take the full advantage of these modes, the H.264 encoder can determine the mode that meets the best RD tradeoff using the RDO mode decision scheme. That is, it exhaustively searches the best mode for every block that produces the minimum rate–distortion cost given by

$$J(s, c, m|QP, \lambda_m) = \text{SSD}(s, c, m|QP) + \lambda_m \cdot R(s, c, m|QP), \quad (1)$$

where QP is the macroblock quantization parameter, λ_m is the Lagrangian multiplier (usually chosen to be $0.85 \cdot 2^{QP/3}$), $\text{SSD}(\cdot)$ means the sum of the squared differences between the original 4×4 luminance block denoted by s and its reconstruction c at each candidate mode m , and $R(\cdot)$ represents the number of bits associated with the chosen mode. The complexity of the Intra-mode decision using RDO is extremely high. To reduce the encoding complexity, a new Intra-mode decision method is proposed using a multistage decision framework [13] as detailed below.

The RDO mode decision method finds the optimal mode in the RD sense. To be more precise, it exhaustively searches the best mode by measuring the RD cost based on the actual rate and distortion after entropy coding and reconstruction, respectively. This technique is however too costly to be implemented in practical applications. It is desirable to find the optimal mode or a sub-optimal mode using some simplified method [8].

3. Feature selection

In this section, we consider both spatial and frequency domain features to filter out unlikely mode candidates.

3.1. Spatial domain feature selection

For each prediction mode, we can compute the sum of absolute differences (SAD) between the true and the predicted pixel values as a spatial domain feature. It can be written as

$$\text{SAD} = \sum_{(x,y) \in b_k} |D(x,y)|, \quad D(x,y) = I(x,y) - P_i(x,y), \quad (2)$$

where b_k represents k^{th} 4×4 block in current macroblock, I and P_i represent the true and the predicted pixel values of i^{th} 4×4 Intra mode, respectively.

Intuitively speaking, a good mode should lead to a small SAD value. This is called the amplitude test. It is important to study the accuracy of mode selection based on this simple test. To achieve this goal, one way is to examine the probability that a wrong mode is chosen for the current block, given observation of $f(m_{\text{SAD}})$. It is referred to as the a posteriori error probability since it contains the information after $f(m_{\text{SAD}})$ is observed

$$\text{Prob}(e|f(m_{\text{SAD}})) = \text{Prob}(m_{\text{SAD}} \neq m_{\text{RDO}}|f(m_{\text{SAD}})), \quad (3)$$

where m_{SAD} and m_{RDO} are modes selected based on the optimal SAD and RDO criteria, respectively, and $f(m_{\text{SAD}})$ represents the SAD value of mode m_{SAD} . Intuitively speaking, the smaller the value of $f(m_{\text{SAD}})$, the smaller the probability of erroneous prediction. This conjecture is studied experimentally in Fig. 1A, where we plot $\text{Prob}(e|f(m_{\text{SAD}}))$ as a function of $f(m_{\text{SAD}})$. The statistical data are obtained from a collection of three QCIF sequences (Akiyo, Foreman, and Table Tennis) of 300 frames long. We see that if the smallest SAD value, i.e., $f(m_{\text{SAD}})$ is lower than 50, we can get an excellent prediction with the error probability lower than 10%. For most of bigger SAD values than 50, the prediction error is approximately in the

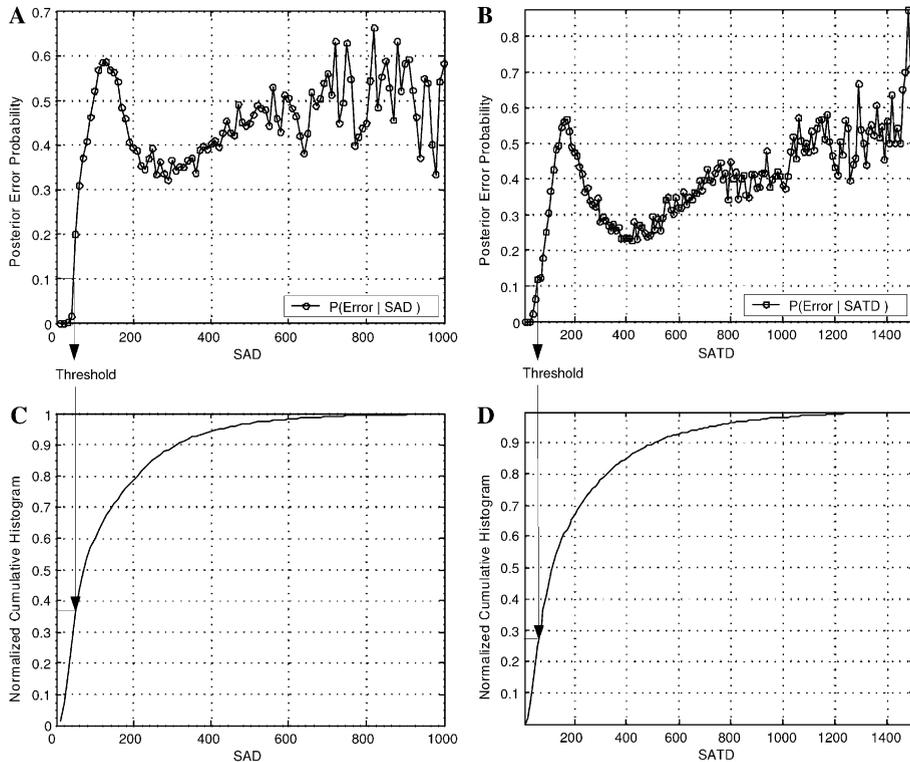


Fig. 1. The posterior error probability calculated based on (A) the definition in (3), (B) the definition in (5), (C) the cumulative histogram with respect to the SAD value, and (D) the cumulative histogram with respect to the SATD value.

range between 30 and 60%. Then, the prediction based on the smallest SAD value alone is not very accurate.

The posterior error probability figure shows the possibility of early termination with sacrifice of decision accuracy. Based on experimental results, the RD performance loss is insignificant in most test sequences when the decision error is less than 10%. Due to this reason, the proposed algorithm employs the early termination rule using the SAD feature given in (3). That is, if the $f(m_{\text{SAD}})$ value is less than threshold $\Delta_{\text{SAD}} = 50$, then we can choose m_{SAD} as the Intra-prediction mode. The usefulness of this early termination rule can be found by examining the cumulative histogram of blocks with computed feature values of $f(m_{\text{SAD}})$. For example, under threshold $\Delta_{\text{SAD}} = 50$, there are about 38% of blocks in Fig. 1C that have $f(m_{\text{SAD}}) \leq \Delta_{\text{SAD}}$ for early termination at the risk of less than 10% in making the wrong mode selection.

It is worthwhile to point out that the spatial edge filtering technique proposed in [8] passes the examining block only once while the SAD and SATD computations have to be repeated for every possible Intra-mode in our scheme. However, the edge filtering method always selects the four most probable modes for RDO since it does not have enough accuracy in picking up one or two final modes. The complexity of H.264 Intra-mode decision lies primarily in RDO or RD estimation rather than the simple feature calculation like SAD. Our algorithm has focused on the reduction of the number of RDO and, consequently, achieved a modular speedup of around 10.

3.2. Transform domain feature selection

Based on the Parseval theorem, the total energy (or the 2-norm) of the difference in the space domain and the transform domain should be the same. It is reasonable to say that a good prediction should also produce a small value of the sum of the absolute transform coefficient differences (SATD), which is defined by

$$\text{SATD} = \sum_{(x,y) \in b_k} |T\{D(x,y)\}|, \quad (4)$$

where $D(x,y)$ is defined in (2) and T is a certain 2D orthonormal transform. To compute the SATD feature, T is chosen to be the separable four-point Hadamard transform along each dimension due to its simplicity and good performance in assisting the mode selection. Note that the Hadamard transform can be implemented with only addition and shift operations that are computationally efficient. In fact, it is observed that the computation of SATD does not increase the overall computational cost much. Also, it is important to point out that SAD and SATD are generally different since they are the 1-norms (rather than the 2-norms) of spatial- and transform-domain signals, respectively.

Following the discussion for SAD, we can analyze the usefulness of the SATD feature by examining the posterior probability of erroneous decision. Let us consider the posterior error probabilities based on SATD observation below.

$$\text{Prob}(e|f(m_{\text{SATD}})) = \text{Prob}(m_{\text{SATD}} \neq m_{\text{RDO}}|f(m_{\text{SATD}})), \quad (5)$$

where m_{SATD} and m_{RDO} denote the best SATD mode and the best RDO mode, respectively, and $f(m_{\text{SATD}})$ denotes SATD value for mode m_{SATD} . The posterior error probability is plotted in Fig. 1B and the cumulative histogram of blocks with respect to the SATD is shown in Fig. 1D. The statistical data are sampled from the same sequences with SAD feature like Akiyo, Foreman, and Table Tennis QCIF sequences of 300 frames. Similarly, we can derive the early termination rule based on the SATD feature. In the next section, we will show that it is advantageous to use both SAD and SATD features jointly.

3.3. Sequential mode filtering

We can apply the selected features one by one in cascade to filter out unlikely modes (or to reach an early termination criterion). This is called the sequential filtering process. The effectiveness of the sequential filtering process generally depends on the order of features. According to the above discussion, we have two features, i.e., SAD and SATD. We do not have a conclusion about which feature should go first to guarantee a better filtering performance. However, for the complexity concern, it is worthwhile to go with the SAD feature first. The amplitude tests for the SAD feature as stated in Section 3.1 can be applied. Then, we can move to the SATD feature. It is worthwhile to point out that there exist strong correlations between SAD and SATD features. Thus, it is difficult to get an effective filtering result by cascading them in two stages. In the next section, we will develop a better mode filtering scheme that uses the SAD and SATD features jointly. By using the joint mode filtering scheme, we can exploit the mode-priority correlation between spatial and frequency domain features.

4. Mode filtering with joint features

In this section, we will present an approach to filter out unlikely modes with the SAD and SATD features jointly.

4.1. Rank-ordered joint features

Let us rank the computed SAD and SATD values from the smallest to the largest as shown in Fig. 2, where the horizontal and the vertical directions show the rank-ordered SAD and SATD features, respectively. A smaller index number denotes a smaller value. For the case of nine candidate modes, we can obtain a matrix of size 9×9 . However, only 9 locations of the 81 empty slots will be filled by a certain candidate mode. As shown in the figure, mode m_2 has the smallest SAD value and the second smallest SATD value.

The next step is to choose a small screen window to concentrate. In Fig. 2, a screen window of size 3×3 is chosen to cover the three smallest SAD and SATD locations. Modes m_0 , m_2 , and m_4 are located inside the window in this example. This means these modes have both small SAD and SATD values. For the modes outside

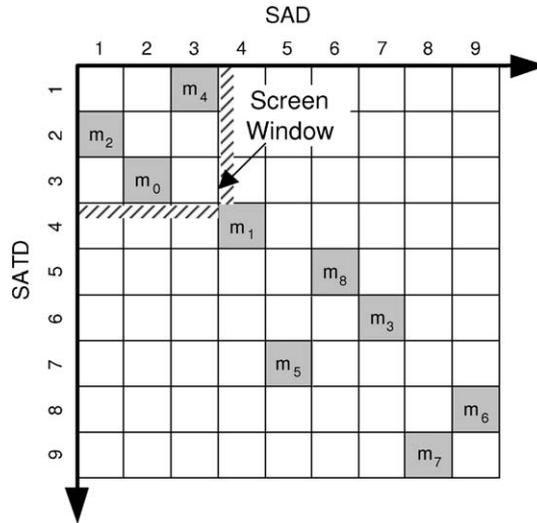


Fig. 2. An example to illustrate the relationship between the 9 candidate modes (m_0, \dots, m_8) and their ranked SAD and SATD values.

the window, they have at least one larger feature value (with its rank equal to 4 or higher).

Since each column or each row can only have one mode, it is easy to see that a screen window of size $K \times K$ can at most contain K modes. A smaller value of K implies a more aggressive filtering scheme. In the next section, we will argue that $K = 3$ is usually a good choice. Then, we will have at most three candidate modes left for further selection.

4.2. Screen window size selection

To justify the screen window size selection, let us examine the distribution of the RDO mode in the rank-ordered joint feature space. Based on the data obtained from two QCIF sequences of 300 frames long, we plot the cumulative histograms of RDO modes in Fig. 3. It is clear that most RDO modes fall in the window of 3×3 lowest ranks. Furthermore, we plot the cumulative histogram of RDO modes as a function of K , where $K = 1, 2, \dots, 9$ is the screen window size in Fig. 4. As shown in the figure, 93–95% of the best mode among the candidates when the 3×3 screen window is selected. On the other hand, increasing the window size more than 3×3 does not improve the overall RD performance much but demands one more additional RDO process. Thus, we can efficiently search the optimal mode by focusing on modes that fall in this region.

The other way to justify the screen window size selection is to perform the rate–distortion analysis. Given a specific rank pair (s_1, s_2) in the joint SAD–SATD feature domain. We can compute the average rate and distortion increase with respect to the RDO mode as

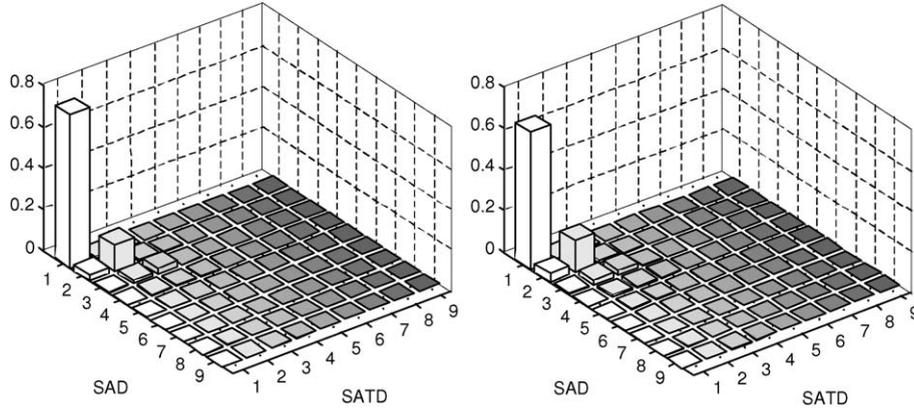


Fig. 3. The histogram of RDO modes in the rank-ordered joint feature space for test sequences “Akiyo” (left) and “Foreman” (right).

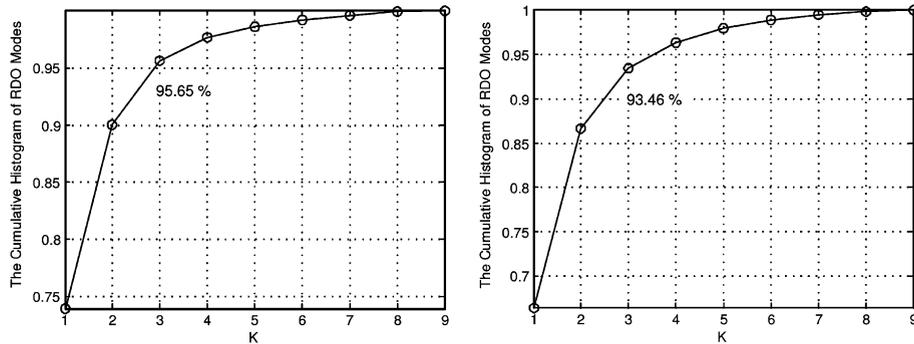


Fig. 4. The cumulative histogram of RDO modes as a function of K , where the screening window is of size $K \times K$, for test sequences “Akiyo” (left) and “Foreman” (right).

$$\Delta R(s_1, s_2) = N_{s_1, s_2}^{-1} \sum_{i \in S_{s_1, s_2}} [R_{s_1, s_2}(i) - R_{\text{RDO}}(i)],$$

$$\Delta D(s_1, s_2) = N_{s_1, s_2}^{-1} \sum_{i \in S_{s_1, s_2}} [D_{s_1, s_2}(i) - D_{\text{RDO}}(i)],$$

where i is the block index, S_{s_1, s_2} is the set of all events in which there exists a Intra-prediction mode in (s_1, s_2) , N_{s_1, s_2} is the cardinality of S_{s_1, s_2} . Specifically, the subscript (s_1, s_2) represents the feature rank pair such as (SAD rank, SATD rank), for instance, mode 0 (DC mode) could be ranked as the second in terms of the SAD cost and the third in terms of SATD. Then, mode 0 falls in the position of (2,3). Further-

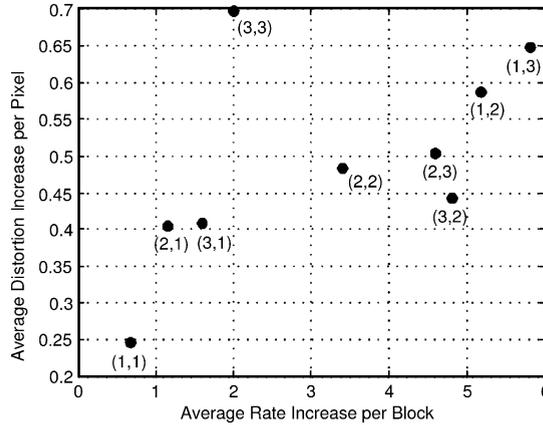


Fig. 5. The average rate (the x -axis) and the average distortion increase (the y -axis) for the QCIF Mobile sequence of 300 frames.

more, S_{s_1, s_2} represents the event that there is a mode falling in the rank pair (s_1, s_2) . If the mode is the true one, it will not result in any rate increase and/or the distortion increase. Otherwise, it will. We use $R_{s_1, s_2}(i)$ and $D_{s_1, s_2}(i)$ to denote the bit rate and the distortion for block i , respectively. Please note that $\Delta R(s_1, s_2)$ and $\Delta D(s_1, s_2)$ can be viewed as the posterior bit rate increase and distortion increase by choosing the mode associated with (s_1, s_2) as the final mode.

We plot $\Delta R(s_1, s_2)$ and $\Delta D(s_1, s_2)$ as the x - and y -axis in Fig. 5 for the QCIF Mobile sequence, where the origin denotes the RDO mode and the modes within the lowest 3×3 rank window are labelled by the rank order. We see from the figure that $(s_1, s_2) = (1, 1)$ is the closest to the RDO mode in the rate–distortion tradeoff. The next closest one is $(2, 1)$, and the third one is $(3, 1)$, and so on. Similar RD performance in the rank-ordered joint feature domain has been observed in other test sequences such as Akiyo, Foreman, and Stefan.

5. Final mode selection

Based on the arguments given in Section 4.2, we conclude that the candidate modes to be considered are those modes that fall in the lowest 3×3 window. The number of Intra-prediction modes in this region is typically 2–3, but no more than 3. In this section, we will focus on the final mode selection.

5.1. Feature-based method

If these 2–3 candidate modes have quite distinct values in SAD and/or SATD, we can apply early termination rules as stated in Section 3 to select the best mode. That is, we have two termination criteria: the amplitude test for the smallest SAD and SATD values. It is often that we can filter out about 30–45% (3 or 4 out of 9) reliably

at the first stage. Then, at the second stage, we may filter out, most likely, less than one more unlikely candidate. Thus, we still have about 4 or 5 modes left for joint rank-ordered mode filtering followed by final selection. However, if the smallest SAD (or SATD) is sufficiently large, we cannot make a robust decision based on the features alone. Then, we will turn to the RDO based mode search as given in the next section.

5.2. RDO-based method

We consider the application of the RDO method to these candidate modes if the feature-based method does not apply. Since the RDO complexity is very high, we propose a new RD model to predict the rate and distortion of 4×4 blocks for further complexity reduction [11]. This RD model is obtained by fitting the curve of observed data and demonstrated to give good performance in simulation. The proposed RD model only works well when the quantization parameter is greater than 16 (i.e., excluding very high rate video) due to the limitation of the accuracy of the model. If the quantization parameter is less or equal to 16, the conventional RD optimization process as described in Section 2 is adopted.

The distortion model is based on the quantization error between coefficients and transformed coefficients of the target block, rather than the quantization parameter [14,15] or the block-variance based quadratic model [16,17], since the latter is not accurate enough for the RD estimation of 4×4 blocks in the experiments. Fig. 6 shows that the relationship between the actual distortion and the estimated distortion. The actual distortion means the distortion between the original and reconstructed blocks. The relationship is approximately linear in the natural logarithmic domain. This leads to the following logarithmic distortion model:

$$\ln(D) = \begin{cases} \mu_1 \cdot \ln(E_Q) + \eta_1 & \text{if } E_Q \leq e^{\lambda_0}, \\ \mu_2 \cdot \ln(E_Q) + \eta_2 & \text{otherwise,} \end{cases}$$

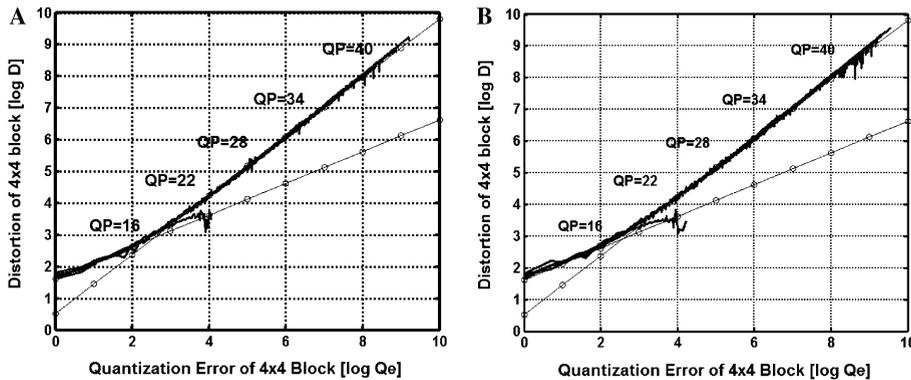


Fig. 6. The two line segments labelled by solid thick lines give the distortion model as a function of the quantization error: (A) Akiyo.QCIF, (B) Stefan.QCIF.

where E_Q is the quantization error, χ_0 is the coordinate at intersection of two lines and μ_k, η_k are parameters to characterize the piecewise linear model as shown in Fig. 6. Therefore, we have

$$D = e^{\mu_k \cdot \ln(E_Q) + \eta_k}, \quad k = 1, 2.$$

With this model, the distortion can be accurately estimated without actual reconstruction.

The actual coding bit rate depends on the entropy coding scheme. For the JVT baseline, the entropy codec is context adaptive variable length code (CAVLC). CAVLC encodes five different types of symbols (or called tokens):

1. The coefficient token (the number of coefficients, the number of trailing ones).
2. The sign of trailing ones.
3. The level of nonzero coefficients.
4. The total number of zeros before the last coefficients.
5. The run of zeros.

Here, we propose a rate model that predicts the rate of a 4×4 block using the above five tokens. Actually, the rate estimation of 4×4 blocks is difficult since H.264 entropy coding is context adaptive [18]. To get an accurate rate model, we tune the parameters carefully so that the estimated rate is close to the actual rate and the performance is consistent from low motion, smooth texture sequence to high motion, complex texture sequence as shown in Fig. 7. First, the rate of the given 4×4 block is estimated by the sum of the bits spent for each token such as

$$\hat{R}_{4 \times 4} = \sum_x C(x) = \sum_x \omega_x \cdot x + \alpha_x,$$

where x is one of five tokens for the given 4×4 block, $C(x)$ is the weighted bit cost function of x (encoding tokens), and ω_x, α_x are corresponding weight and constant for each encoding tokens.

Second, the estimated rate $\hat{R}_{4 \times 4}$ is refined furthermore using the relationship between the true and the estimated rates. The model parameters ω_x and α_x are obtained

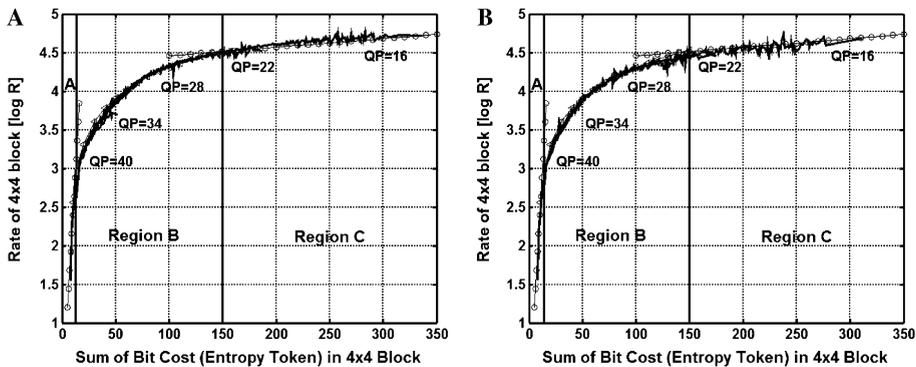


Fig. 7. The predicted rate (circled line) model and actual bit rate (solid curves): (A) Akiyo and (B) Stefan.

empirically based on the observations for the various test sequences. The true bit rate as a function of estimated rate is shown in Fig. 7. The curve fitting method is used to approximate the mapping function which maps estimated rate into true rate. For the simple and accurate curve fitting, proposed rate model is obtained in three different regions:

$$R = \begin{cases} e^{k_1 \cdot \hat{R}_{4 \times 4}}, & \text{Region A} \\ e^{k_2 \cdot \ln(\hat{R}_{4 \times 4} - k_3) + k_4}, & \text{Region B} \\ e^{k_5 \cdot \hat{R}_{4 \times 4} + k_6}, & \text{Region C,} \end{cases}$$

where the parameter vector $K = [k_1, k_2, \dots, k_6]$ in regions A and C are obtained by linear regression and those in region B are obtained by minimizing MSE. Fig. 7 compares real bit rate and the predicted bit rate using the proposed rate model. They are close to each other.

In summary, the proposed algorithm can be described using six major steps as given below.

Feature based fast-Intra-mode decision algorithm

Step 1. SAD Feature: Calculate SAD cost of nine modes for the given 4×4 block based on (2).

Step 2. SAD Early Termination: If the minimum SAD value is less than threshold Δ_{SAD} , which is pre-determined by posterior error probability, then the decision process is terminated with the SAD best mode as the final mode.

Step 3. SATD Feature: Calculate the SATD cost of nine modes for the given 4×4 block based on (4).

Step 4. SATD Early Termination: If the minimum SATD value is less than threshold Δ_{SATD} , then the decision process is terminated with the SATD best mode as the final mode.

Step 5. Unlikely Mode Filtering: Sort the modes in an ascending order of joint features, SAD and SATD values, by using the Quick-sort algorithm [19]. Screen the less likely modes placed outside of 3×3 screen window in the joint rank-order table.

Step 6. RDO Final Decision: We switch the fine level decision to the RDO/RD based on the quantization parameter (QP). If QP is less than or equal to 16, then the RDO method is used to search the final mode. Otherwise, the RD model is used to search it among the most probable modes from Step 5.

6. Experimental results

The mode decision scheme as described in Sections 4 and 5 has been integrated with the H.264 JM7.3a codec for performance evaluation. It is compared with the RDO mode decision of H.264 in terms of the computational cost (the average CPU time per call for the mode decision routine) and the average PSNR as a function of the coding bit-rate for test sequences recommended in [12]. The frame rate of each test sequences is 30 frames/s and the frame skip was selected as five so that total

encoded frames are 50 frames per sequence because total number of frames are 300 frames.

Two different frame formats, QCIF (176 × 144) and D1 (720 × 480), have been examined to verify the performance deviation due to different frame sizes. All macroblocks are intra-coded and the quantization parameter set is chosen to be [10,16,22,28,34,40], which means the step size is doubled from 2 to 64 because H.264 uses exponentially increasing quantization step size scaling scheme. For this set of quantization parameters, the average number of bits per frame, the average PSNR, and the average encoding time of the core mode decision routine are measured for comparison, respectively.

The rate–distortion and complexity performance comparison at various coding rates and frame sizes are shown in Figs. 8–11 and summarized in Tables 1 and 2.

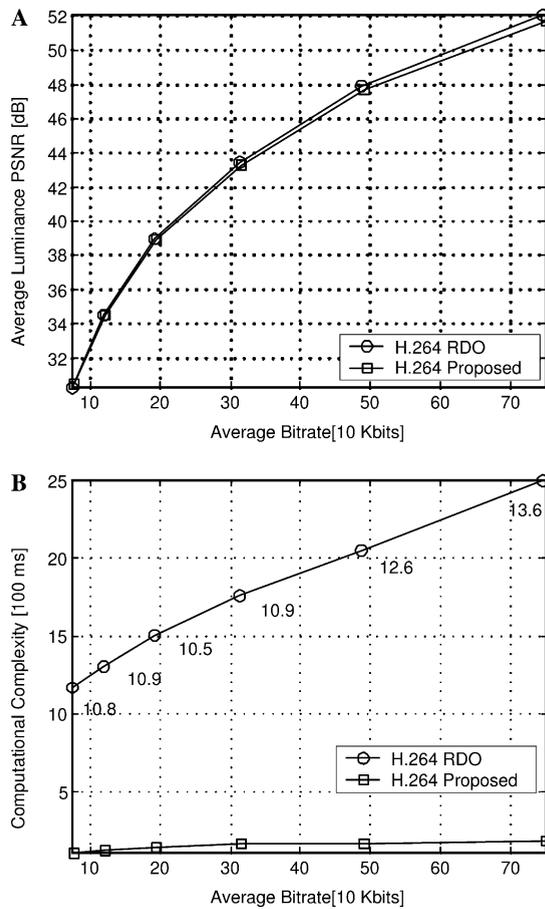


Fig. 8. Comparison of (A) the RD performance and (B) the computational complexity for the Akio QCIF sequence.

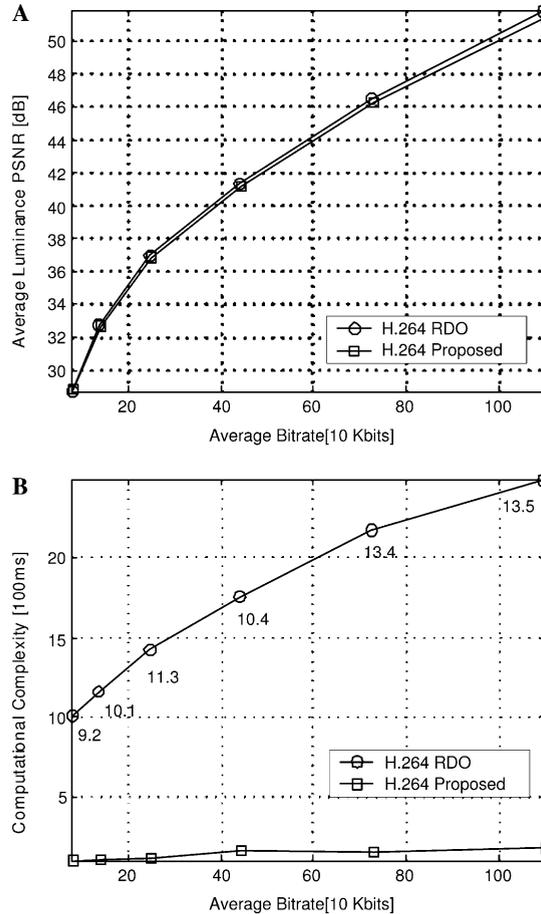


Fig. 9. Comparison of (A) the RD performance and (B) the computational complexity for the Foreman QCIF sequence.

As shown in these tables, the percentage of Intra coding highly varies depending on the input video characteristics such as motion activity and texture complexity. It also fluctuates with the values of coding parameters such as the frame rate. For this reason, we reported both the modular encoding speedup for the Intra-mode decision and the total Intra encoding speedup factor to show the computational savings due to the proposed Intra-mode decision algorithm. The speedup factor is defined to be the ratio of the encoding time of the mode decision routine using the RDO technique and that of the proposed scheme.

The modular time is measured as the average encoding time of the 4×4 Intra-mode decision routine while the total Intra encoding time is measured as the average encoding time of all Intra frames. The encoding time of each method is measured by the function timing profile in Visual Studio.

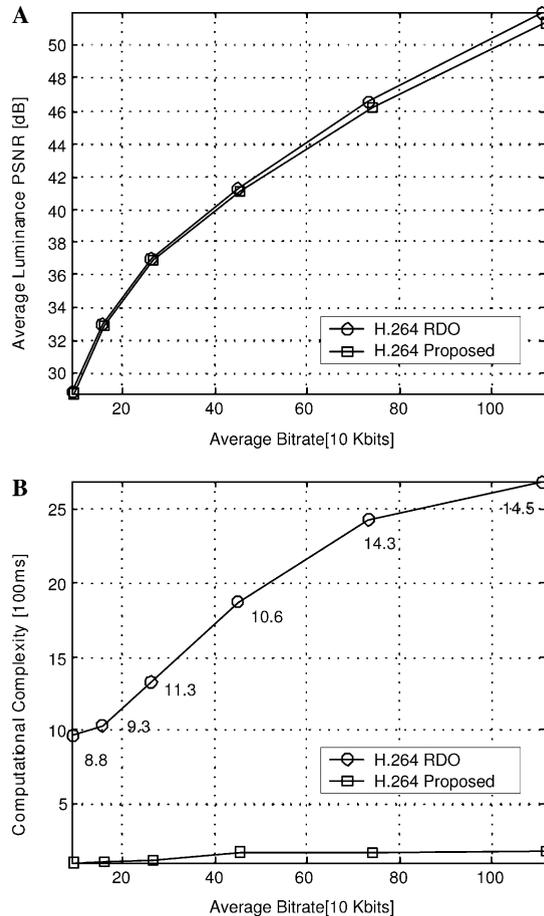


Fig. 10. Comparison of (A) the RD performance and (B) the computational complexity for the Table Tennis QCIF sequence.

Test sequences are chosen from low motion and smooth texture sequence to high motion and complex texture sequence among the MPEG Class A, B, and C sequences in [12]. In terms of the rate–distortion performance, the proposed scheme achieves almost identical RD performance while providing a modular speed-up factor ranging from 8.8 to 14.6 as shown in Figs. 8–11. Note that the computational complexity of the H.264 RDO process increases much faster than the proposed algorithm as the average bit rate increases. The reason is that the RDO procedure consists of both rate and distortion measurements.

When a smaller quantization parameter (QP) is used, the bit rate increases. The RDO procedure consists of rate and distortion measurements that require entropy coding and reconstruction of the current macroblock. The additional residual information from smaller QP demands more computation time of the RDO procedure accordingly. Contrary to the RDO procedure, the proposed algorithm does not

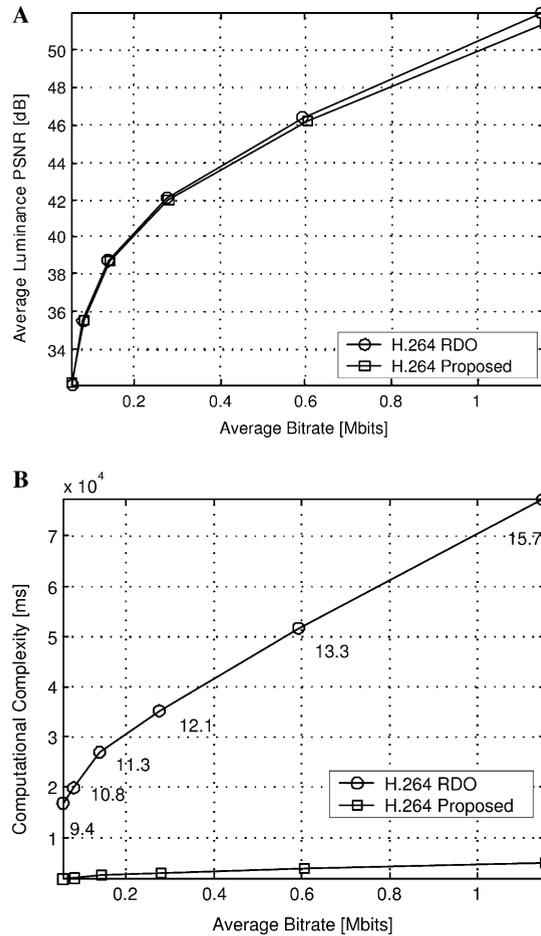


Fig. 11. Comparison of (A) the RD performance and (B) the computational complexity for the Susie D1 sequence.

require entropy coding and reconstruction so that the speedup factor increases as the bit rate increases.

For subjective quality comparison, reconstructed frames based on RDO and the proposed method are captured at three different QPs as shown in Fig. 12. Again, it is difficult to tell the visual quality difference between two pictures obtained by two different mode decision method.

7. Conclusion and future work

The problem of Intra-mode decision for H.264 based on joint features was studied in this research. Two simple features, SAD and SATD, are used to filter out the majority of candidate modes. Only 2–3 modes are left for the final mode decision,

Table 1
Rate and distortion comparison [QP = 10–40]

R:RDO:	Rate (kbit/frame)						PSNR (dB)						
	M:Proposed:	10	16	22	28	34	40	10	16	22	28	34	40
Akio [QCIF]													
R		74.6	48.8	31.2	19.1	11.8	7.4	52.0	47.9	43.4	38.9	34.5	30.3
M		75.1	49.1	31.5	19.4	12.1	7.6	51.7	47.7	43.3	38.9	34.5	30.4
Foreman [QCIF]													
R		109	72.5	43.8	24.8	13.8	8.0	51.8	46.4	41.3	36.9	32.7	28.7
M		109	73.0	44.2	25.0	14.1	8.2	51.3	46.2	41.2	36.8	32.7	28.8
T. T. [QCIF]													
R		110	73.3	44.8	26.1	15.7	9.2	51.9	46.5	41.2	36.9	32.9	28.8
M		111	74.0	45.3	26.6	15.9	9.4	51.3	46.2	41.1	36.8	32.8	28.8
Mobile [QCIF]													
R		208	159	118	81.5	51.6	26.8	52.0	46.4	40.6	34.8	29.2	24.1
M		210	161	119	82.8	52.4	27.4	51.3	45.9	40.1	34.4	28.9	24.0
Susie [D1]													
R		1147	595	275	141	81.5	56.5	51.9	46.3	42.1	38.7	35.5	32.1
M		1153	606	280	146	84.3	58.4	51.4	46.2	42.0	38.7	35.6	32.3
Mobile [D1]													
R		2334	1703	1153	740	455	247	51.9	46.3	40.7	35.6	30.6	25.9
M		2348	1717	1165	750	462	253	51.4	45.9	40.4	35.4	30.4	25.8

Table 2
Computational complexity comparison [QP = 10–40]

Test sequences		Modular speedup (scale)						Total speedup (scale)					
		10	16	22	28	34	40	10	16	22	28	34	40
Akio	QCIF	13.6	12.6	10.9	10.5	10.9	10.8	6.7	6.9	6.6	7.4	7.1	5.7
Foreman	QCIF	13.5	13.4	10.4	11.3	10.1	9.2	6.4	7.3	7.1	7.0	7.1	6.7
T. T.	QCIF	14.5	14.3	10.6	11.3	9.3	8.8	6.2	7.0	6.8	7.7	7.2	6.8
Mobile	QCIF	14.3	14.8	12.1	15.4	14.3	11.9	5.8	7.4	7.5	7.0	6.8	6.7
Susie	D1	15.7	13.3	12.1	11.3	10.8	9.4	7.5	7.2	6.4	7.3	6.8	6.5
Mobile	D1	12.9	12.9	12.3	12.5	12.1	11.0	7.4	7.2	7.1	6.5	6.3	5.7

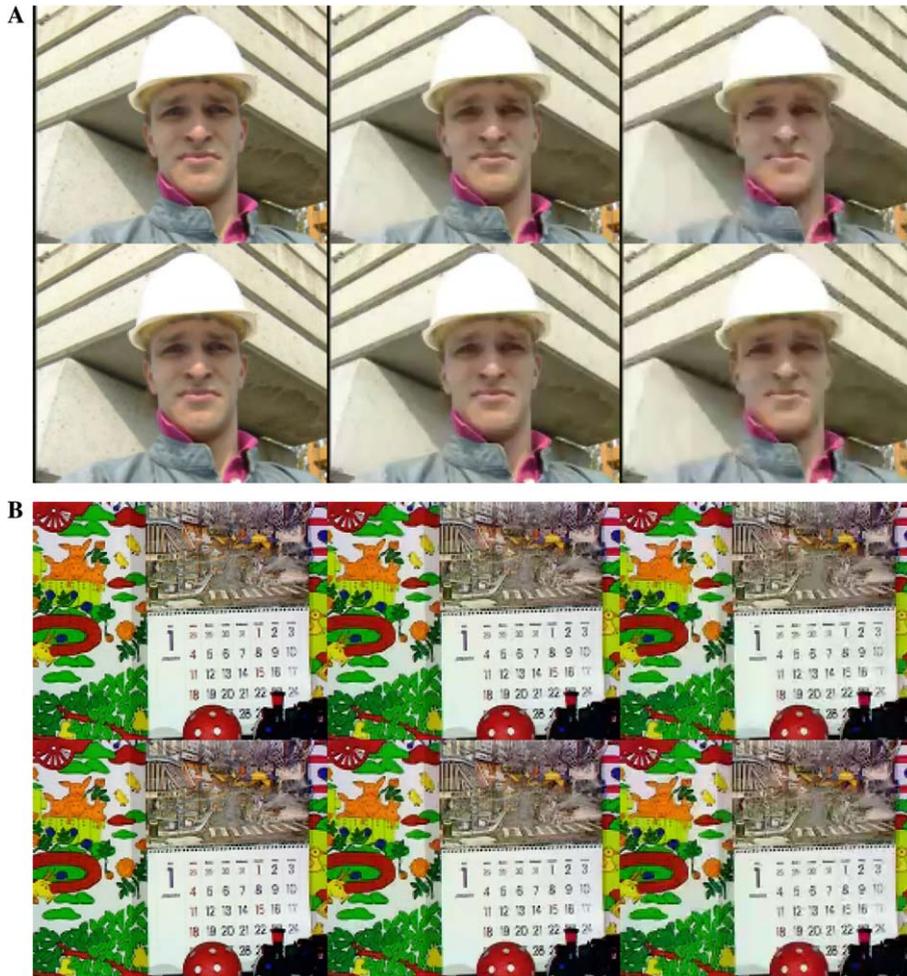


Fig. 12. Visual quality comparison of reconstructed QCIF frames of (A) Foreman and (B) Mobile using the RDO method (the upper row) and the proposed method (the lower row) where $QP = 16, 28,$ and 34 for the left, middle, and right columns, respectively.

which can be done by feature-based or RDO-based methods. It was demonstrated by experimental results shows that we can speed up the Intra prediction module of the JVT reference software JM7.3a by a factor of 10 times or more without noticeable RD performance degradation. Future research topics include the low complexity Intra/Inter-mode decision for the H.264 which is the natural extension of this research.

References

- [1] UB video, Emerging H.26L standard: overview and TMS320C64x digital media platform implementation, White paper, UB video Inc., Feb 2002.

- [2] P. Topiwala, G. Sullivan, A. Joch, F. Kossentini, "Performance evaluation of H.26L TML 8 vs. H.263++ and MPEG4," Document VCEG-O42, 15th Meeting, ITU-T Q.6/SG16, Video Coding Experts Group, Pattaya, Thailand, December 2001.
- [3] B. Erol, M. Gallant, G. Cote, F. Kossentini, "The H.263+ video coding standard: complexity and performance," in: IEEE Data Compression Conference, Snowbird, Utah, March 1998, pp. 259–268.
- [4] Xiang Li, Guowei Wu, Fast integer pixel motion estimation, JVT-F011, 6th Meeting, Awaji Island, Japan, December 5–13, 2002.
- [5] Hye-Yeon Cheong, Alexis M. Tourapis, Pankaj Topiwala, Fast motion estimation within the JVT Codec, JVT-E023, 5th Meeting, Geneva, Switzerland, October 9–17, 2002.
- [6] T. Halbach, "Performance Comparison: H.26L Intra Coding vs. JPEG2000," ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, JVT 4th Meeting Klagenfurt, Austria, July 2002.
- [7] T. Wiegand, G.J. Sullivan, G. Bjontegaard, A. Luthra, Overview of the H.264/AVC video coding standard, IEEE Trans. Circuits Syst. Video Technol. 7 (2003) 1–19.
- [8] F. Pan, X. Lin, et. al., Fast Mode Decision for Intra Prediction, ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, JVT 7th Meeting Pattaya II, Thailand, March 2003.
- [9] B. Jeon, J. Lee, Fast Mode Decision for H.264, ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, JVT 10th Meeting, Waikoloa, Hawaii, December, 2003.
- [10] Changsung Kim, Hsuan-Huei Shih, C.C. Jay Kuo, Feature-based intra-prediction mode decision for H.264, in: IEEE Proceedings of International Conference Image Processing, Singapore, vol. 2, October, 2004, pp. 769–772.
- [11] Changsung Kim, Qing Li, C.C. Jay Kuo, Fast Intra-prediction model selection for H.264 codec, in: SPIE International Symposium ITCOM 2003, Orlando, FL, July, 2003.
- [12] JVT Test Model Ad Hoc Group, Evaluation Sheet for Motion Estimation, ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, Draft version 4, February, 2003.
- [13] Changsung Kim, Hsuan-Huei Shih, C.C. Jay Kuo, Multistage Mode Decision for Intra Prediction in H.264 Codec, in: IS&T/SPIE 16th Annual Symposium EI, Visual Communications and Image Processing, Orlando, FL, January, 2004.
- [14] K.H. Yang, A. Jacquin, N.S. Jayant, A normalized rate–distortion model for H.263-compatible codecs and its application to quantizer selection, in: IEEE Proceedings of International Conference Image Processing, vol. 2, October, 1997, pp. 41–44.
- [15] K.H. Yang, Arnaud Jacquin, Real time implementation of rate–distortion optimized coding mode selection for H.263 video coders, in: IEEE Proceedings of International Conference Image Processing, vol. 2, October, 1998, pp. 387–390.
- [16] Tihao Chiang, Y.Q. Zhang, A new rate control scheme using quadratic rate distortion model, IEEE Trans. Circuits Syst. Video Technol. (1997).
- [17] Hwangjun Song, Jongwon Kim, C.-C. Jay Kuo, Hybrid DCT/wavelet I-frame coding for efficient H.263+ rate control at low bit rate, in: IEEE International Symposium on Circuits and Systems, Monterey, California, May 31–June 3, 1998.
- [18] Zhihai He, Sanjit K. Mitra, A unified rate distortion analysis framework for transform coding, IEEE Trans. Circuits Syst. Video Technol. 11(12) (2001).
- [19] T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, Introduction to algorithms, second ed., The MIT Press and McGraw-Hill Book Company, 2001.