Feature-Based Intra-/InterCoding Mode Selection for H.264/AVC

Changsung Kim, *Member, IEEE*, and C.-C. Jay Kuo, *Fellow, IEEE*

***Abstract*—A fast feature-based intra/inter coding mode selection scheme for the H.264/AVC video coding standard is proposed in this paper. First, three features are extracted from a macroblock to form a feature space. Then, the feature space is partitioned into three regions, i.e., risk-free, risk-tolerable, and risk-intolerable regions, where the risk is calculated using the rate-distortion (RD) performance loss due to wrong mode decision as well as the probability distribution of inter/intra modes in the feature space obtained from an off-line training process. Depending on the region where the feature vector of a macroblock is located, we can apply mechanisms of different complexity for final mode decision. To calculate the likelihood function of the risk, both parametric and nonparametric density estimation schemes are developed to offer different rate-distortion–complexity tradeoffs. It is demonstrated by experimental results that the proposed algorithm can save approximately 20%–32% of the total encoding time of H.264 (JM7.3a) with little degradation in the rate-distortion performance.**

***Index Terms*—Bayes-risk minimization, coding mode prediction, density estimation, encoder complexity, H264/AVC, mode decision.**

I. Introduction

THE H.264/MPEG-4 Part 10 AVC (or abbreviated as H.264/AVC) is an emerging video coding standard jointly developed by ITU-T and MPEG [1], [2]. It has been developed to enhance coding efficiency to meet the increasing demand for high-quality multimedia contents and services. H.264/AVC has improved the coding gain of previous standards over a wide range of bit rates by allowing a rich set of coding modes. Generally speaking, the rate-distortion (RD) performance can be optimized by choosing the mode whose Lagrangian RD cost is the minimal. However, the selection of these optimal modes is nontrivial, which usually demands a large amount of computation. Thus, complexity reduction for H.264 encoding has been an active research area. The objective is to reduce the encoding complexity while keeping the RD performance as close to that of full search as possible.

We focus on the problem of encoder complexity reduction in this paper by considering a coarse-level binary coding mode prediction, i.e., whether the intra or the inter predictive mode should be adopted for a given macroblock of P- or B-frames. Recently, there has been a lot of research on fast intra mode decision [3], [4] and fast inter mode decision [5]–[7]. However, few results have been reported in the area of intra/inter coding mode prediction. Chen *et al.* [8] examined a model-based intra/inter mode selection scheme based on simple features, where the costs of intra- and inter-coding were modeled by the variance and the sum of absolute differences (SAD) of a macroblock, respectively. The information of motion vectors and quantization parameters was sometimes taken into account in [8]. Turaga and Chen [9] developed a classification-based mode-decision scheme using the maximum-likelihood (ML) criterion to facilitate video transmission over networks. The above two schemes are however not suitable for intra/inter mode decision in the H.264 reference code [10] since their selected features are not accurate enough to provide efficient mode prediction.

Jagmohan and Ratakonda [11] proposed a supervised binary mode classification scheme using a decision tree. They used the down-sampled sum of absolute transform differences (SATD) values to form a feature space and optimally partitioned it by minimizing the misclassification rate. However, the impact of the RD performance loss incurred by the wrong decision was not considered in their work. More recently, the simple feature-based mode decision algorithm was proposed in [12]. Simply speaking, it always performs the inter mode decision first and checks whether one should perform the intra mode search by comparing the temporal correlation (the average rate of prediction residuals) and the spatial correlation (the sum of boundary pixel errors). This algorithm works well when the inter mode is most likely to be the dominant one. However, a noticeable RD performance loss could be caused by an erroneous decision when the feature difference is not accurate enough. It is worthwhile to emphasize that a wrong decision may or may not be critical depending on the resultant RD performance loss.

A fast inter/intra mode-prediction scheme with carefully selected features is developed in this study. Under this framework, we divide the mode decision into two stages: "coarse-scale decision" and "fine-scale decision." At the first stage, we perform a binary decision to choose either the inter or the intra prediction type to be used for a target block. The objective is to reduce the computational complexity by deciding the most probable intra/inter type earlier. Obviously, there is a risk in making the wrong prediction in the computation-saving effort. Thus, the management of the prediction risk, which is quantified by the averaged RD performance degradation rather than by the misclassification rate alone in this work, is critical. The proposed algorithm adopts three simple features to estimate the temporal correlation, spatial correlation and motion activity, respectively, and then performs the binary mode decision in the 3-D feature space. The 3-D feature vector space is further partitioned into

Manuscript received June 15, 2005; revised March 26, 2006 and October 10, 2006. The work was supported in part by the Integrated Media Systems Center, a National Science Foundation Engineering Research Center, under Cooperative Agreement EEC-9529152. This paper was recommended by Editor-in-Chief C. W. Chen.

C. Kim is with QUALCOMM MediaFLO Technologies, San Diego, CA 92121-1714 USA (e-mail: cskim@qualcomm.com).

C.-C. J. Kuo is with the Department of Electrical Engineering and Integrated Media Systems Center, University of Southern California, Los Angeles, CA 90089-2564 USA (e-mail: cckuo@sipi.usc.edu).

Digital Object Identifier 10.1109/TCSVT.2006.888829

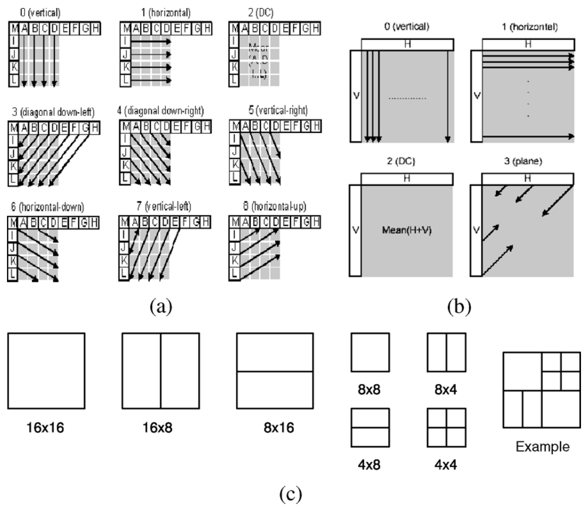


Fig. 1. Intra/inter prediction modes for macroblock: (a) 4×4 intra modes; (b) 16×16 intra modes; and (c) inter modes.

three regions according to the expected RD loss: the risk-free, the risk-tolerable, and the risk-intolerable regions. We apply algorithms of different complexity for final coding mode decision to blocks located in these three regions, which will be described in Section II-B. Once the intra/inter prediction is determined, we proceed to the mode decision at the second stage; namely, which specific intra or inter mode to be used.

The remainder of this paper is organized as follows. The background of intra/inter mode decision is reviewed and an overview of the proposed algorithm is presented in Section II. Feature selection and feature space partitioning are described in Sections III and IV, respectively. Algorithms to predict the inter/intra coding mode in the risk-free and the risk-tolerable regions are described in Sections V-A and V-B. The likelihood estimation for risk-minimized decision is addressed in Section VI. Experimental results are presented in Section VII to compare the proposed algorithm with the rate-distortion optimization method of the H.264 reference code. Concluding remarks and future research topics are given in Section VIII.

II. Background and Overview

A. Intra/Inter Mode Decision Algorithm

The intra/inter mode decision scheme employed in the H.264 reference code is reviewed in this section. It exploits spatial and temporal correlations of underlying video to perform intra/inter mode prediction [13]. As defined in H.264/AVC, there are four categories of predictive coding mode: the skip mode, the direct mode (B-slice), the intra and inter modes. Specifically, H.264 offers four intra modes for 16×16 luma blocks and nine prediction modes for 4×4 luma blocks, as shown in Fig. 1(a) and (b). For the inter mode prediction, seven modes of different sizes and shapes are supported by H.264 as shown in Fig. 1(c).

The H.264 encoder may test all possible modes, which is called the full mode decision, by evaluating the cost associated with each of intra/inter modes and then select the best mode with the smallest cost. The cost is usually defined to be Lagrangian RD cost. To estimate this cost, the H.264 reference code employs an RD optimization (RDO) procedure as stated below.

• Initialization: Given the last decoded frame, the macroblock quantization parameter QP and the Lagrangian multiplier can be selected accordingly.

• Step 1: Calculate the residuals of various intra/inter prediction modes. For inter-predictive modes, perform the motion estimation within a search range for multiple reference frames. For intra-predictive modes, the directional prediction is applied to calculate the residual for each mode.

• Step 2: Select the best prediction mode among all possible intra/inter predictive modes by minimizing the following Lagrangian functional:

$$J(s, c, \mathrm{mode}|QP, \lambda_{\mathrm{mode}}) = SSD(s, c, \mathrm{mode}|QP) + \lambda_{\mathrm{mode}} \cdot R(s, c, \mathrm{mode}|QP) \quad (1)$$

where QP is the quantization parameter, λ_{mode} is the Lagrange multiplier for mode decision, SSD is the sum of the squared differences between the original block luminance (denoted by s) and its reconstruction c, and $R(s, c, mode|QP)$ represents the number of bits associated with the chosen *mode*. It includes the bits needed for the coding of the selected prediction mode and the DCT coefficients for the given block.

The computational cost of the above RDO procedure for intra/inter mode decision is very high. Even without RDO, the complexity is still high since the encoder checks all inter and intra modes. Especially, the RDO procedure for inter modes is more complex than that for intra modes since the former involves full motion search over a window of reference positions. When the best mode is one of the intra modes, a large amount of computation for motion search is wasted.

There exist fast algorithms to select the optimal inter-prediction mode [14]–[16] and the optimal intra-prediction mode [3], [12], [17]–[20], individually. However, little work has been done yet in developing a fast algorithm that performs fast (binary) intra/inter mode decision. It is desirable to make the coarse-level mode decision about which class of modes (i.e., the class of intra-predicted modes or the class of inter-predicted modes) to focus at the first stage. Afterwards, fast algorithms can be used to select the specific mode within each class at the second stage for fine-level mode decision. Our study focuses on the coarse-level mode decision at the first stage.

Even though some of this work was presented in [21] and [22], we have rewritten the material and included a substantial amount of new results. They include statistical analysis of the macroblock distribution in each of partitioned observation feature-vector space, new decision schemes of quantized cells in the 3-D feature vector space, and comparison of parametric and nonparametric density estimation methods in terms of the RD performance and complexity saving for various frame-skips and different resolutions of test sequences (e.g., QCIF, CIF, and D1).

B. Overview of Proposed Algorithm

An overview of the proposed algorithm is depicted in Fig. 2. First, three cost-effective features are extracted from the current macroblock to form a 3-D feature vector. It is observed that the inter prediction provides better performance than the intra prediction when the temporal correlation is stronger than

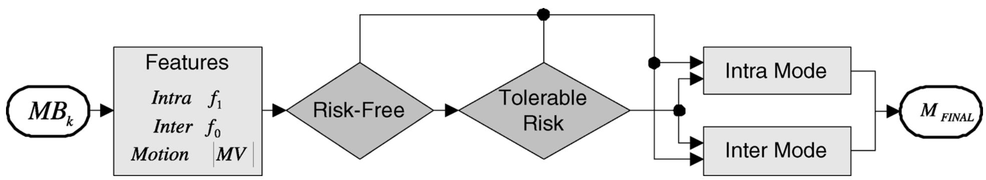


Fig. 2. Overview of the proposed intra/inter mode decision algorithm.

the spatial correlation, and vice versa. We also observe that the intra/inter mode decision is highly correlated with the degree of motion activity. Based on these observations, we employ three features that reflect the spatial correlation, the temporal correlation, and the motion activity, respectively. Second, the feature space is partitioned into three mutually exclusive regions off-line according to the risk, namely, risk-free, risk-tolerable, and risk-intolerable regions.

As shown in Fig. 2, if the feature vector lies in the risk-free region, the decision is made based on simple feature comparison which is the differential cost between the intra feature and the inter feature. If it is in the risk-tolerable region, the risk-minimizing mode is selected, where the risk is chosen to be the Bayes risk, which is calculated based on the probability of erroneous mode selection and the average RD performance loss. Finally, if it is in the risk-intolerable region, i.e., it is included neither in the risk-free region nor in the risk-tolerable one, a full mode decision process is conducted to avoid significant RD loss.

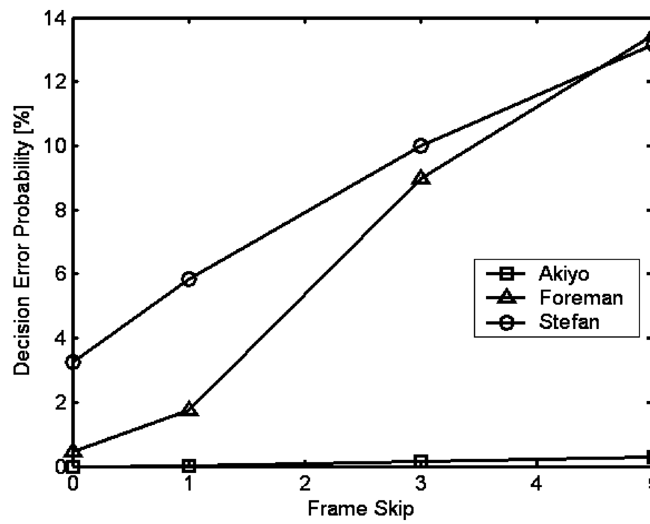


Fig. 3. Decision error probability versus the frame-skip number for sequences Akiyo, Foreman, and Stefan.

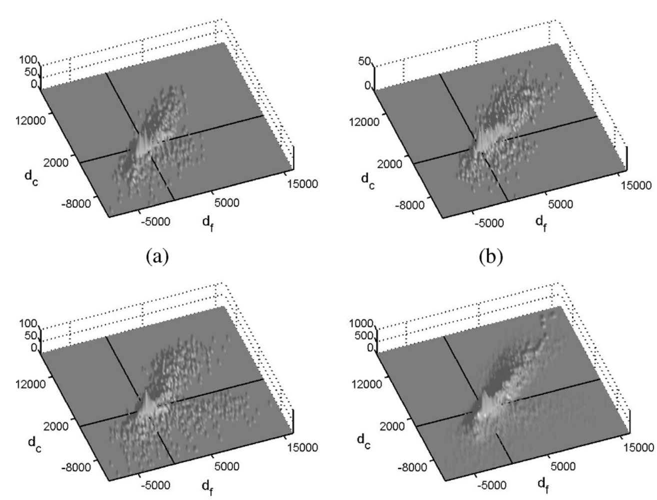


Fig. 4. Cumulative histogram of macroblock distribution using (a) macroblocks of low motion activities, (b) macroblocks of medium motion activities, (c) macroblocks of high motion, and (d) all macroblocks activities.

III. Feature Selection

The proposed mode decision algorithm is a feature-based approach. Good features should be easily computed at a low computational cost while capturing the spatial and/or temporal correlation well so as to offer important clues about which mode to select.

A. Intra Mode Feature

To characterize the spatial domain correlation, which can be exploited by an intra-prediction mode, we use the SATD of the prediction residual due to its simplicity and good mode-discriminating capability [23]. For the transform function, we adopt the simple Hadamard transform since only addition and shift operations are needed in the computation. For each 4×4 block, we calculate SATD values for only five modes, which are the dc, vertical, horizontal, diagonal down-left, and diagonal down-right modes, and pick the smallest one as the representative value. Finally, the feature to reflect the spatial domain correlation, denoted by f_{Intra} or f_1, is chosen to be the sum of the SATD values of sixteen 4×4 blocks contained by the current 16×16 macroblock. Mathematically, we have

$$f_{\text{Intra}} = f_1 = \sum_{k=1}^{16} \text{SATD}(m_k),$$
$$m_k = \underset{m=\{m_0,\ldots,m_4\}}{\operatorname{argmin}} \text{SATD}(m),$$
$$\text{SATD}(m) = \sum_{(x,y)\in B_k} |T(I(x,y) - P_m(x,y))|) \qquad (2)$$

where k is the index of 4×4 blocks B_k in the current macroblock, m denotes one of five candidate prediction modes, $T(\cdot)$ is the Hadamard transform, $I(x,y)$ and $P(x,y)$ are pixels of the kth block of the current macroblock and the corresponding predictive intra mode, respectively. If the value of f_1 is small, it is likely that the intra-predictive mode will be selected.

In calculating intra feature f_{Intra}, the original pixels of its neighbor 4×4 blocks are used to generate the prediction pixels $P(x,y)$. Basically, the intra and inter features are chosen to be simple yet effective for the binary mode decision. At the same time, the reconstruction cost of each 4×4 block is saved in our scheme. We see that the proposed feature difference is very accurate as shown in Figs. 3 and 4 when the MB is in the risk-free region for various frame-skips as well as a wide range of motion activity. By the frame-skip, we mean the number of frames skipped between two encoded frames. For a sequence of 30 fps, we will have 30, 15, and 10 fps if the frame-skip number is equal to 0, 1, and 2, respectively. It is important to emphasize that the main purpose of simple intra and inter features are not for the final intra-mode decision but for simple binary decision of the risk-free region in the observation space.

B. Inter Mode Feature

To compute the feature to reflect the temporal domain correlation, which can be exploited by an inter-prediction mode, we search the best matched macroblock with respect to one reference frame. In our implementation, the motion vector is obtained using modified MVFAST [24] with the quarter pixel accuracy. The MVFAST algorithm can be summarized as follows.

1) Detection of stationary (zero-motion) blocks;
2) Determination of local motion activity;
3) Determination of the search center depends on the local motion activity;
4) Local motion search around the search center.

Three modifications are made to fine-tune the performance. First, we add two more candidate motion vectors in the spatial (the left upper macroblock) and temporal (the co-located macroblock in the previous frame) neighborhood. Second, the residuals of previously visited search points are kept in the memory to avoid recalculation. Third, the total number of search points for a block is restricted to be $M = 512$ in the worst case. Mathematically, the inter-prediction feature, denoted by f_{Inter} or f_0, can be expressed as

$$f_{Inter} = f_0 = \sum_{(x,y)\in MB_l} |T(I(x,y) - Q(x',y'))|, \quad (x',y') = (x,y) + (v_x, v_y) \tag{3}$$

where (v_x, v_y) is the motion vector obtained by the fast search algorithm, and $I(x,y)$ and $Q(x',y')$ are pixels of the current macroblock MB_l and its predictive macroblock.

C. Motion Activity Classification

The third feature used is the magnitude of the motion vector, which can be computed as

$$|MV| = (v_x^2 + v_y^2)^{1/2}. \tag{4}$$

The reason to include the motion vector magnitude (or strength) is that it is related to the reliability of inter-prediction feature f_0. In Fig. 3, we show the decision error probability as we adjust the frame-skip number for three typical test sequences that have different motion activities, i.e., the motion activities of Akiyo, Foreman, and Stefan are low, medium, and high, respectively. The decision metric is chosen to be feature difference as given by

$$d_f = f_{\text{Intra}} - f_{\text{Inter}}. \tag{5}$$

In (5), the intra feature is the sum of SATD values for sixteen 4×4 blocks within a target macroblock of size 16×16 as shown in (2). Also, the inter feature is the SATD value of the macroblock as given in (3). Simply speaking, they are the SATD values of the spatial and the temporal prediction residuals of the same macroblock. Please note that our feature difference is very accurate (with a decision error of less than 6% for three test sequences) when the frame skip is zero (30 fps) or one (15 fps), as shown in Fig. 3.

According to the feature difference measure, the decision error probability can be defined as

$$P(e) = P(d_f < 0|\text{inter}) \cdot P(\text{inter}) + P(d_f > 0|\text{intra}) \cdot P(\text{intra}) \tag{6}$$

where $P(\text{intra})$ (or $P(\text{inter})$) is the probability that best mode is intra (or inter) coding mode. As shown in Fig. 3, we see that the probability of erroneous decision increases as the motion activity grows.

Furthermore, we collect the statistics of macroblocks from seven test sequences (QCIF) with the same quantization parameter, and draw the probability distribution of the RD cost difference d_c and the feature difference d_f in Fig. 4. The RD-cost difference d_c is defined to be the cost difference of the RD function between the best inter mode and the best intra mode written. It can be written as

$$d_c = (D_{\text{Intra}} + \lambda_{\text{Intra}} \cdot R_{\text{Intra}}) - (D_{\text{Inter}} + \lambda_{\text{Inter}} \cdot R_{\text{Inter}}) \tag{7}$$

where λ_{Intra} and λ_{Inter} are the Lagrangian multipliers used in the H.264 reference code. It is easy to verify that if the best mode is an inter-predictive (or an intra-predictive) mode, then d_c is positive (or negative).

We see that the feature difference has excellent correlation with the RD-cost difference as shown in Fig. 4(a). Thus, we may use the feature difference to do the mode prediction, and the overall decision accuracy is 84.7%. However, it is observed that the prediction accuracy degrades as motion activity increases, as shown in Fig. 4(b)–(d). It is also worthwhile to point out that the number of intra modes used for prediction increases as the motion becomes faster. From these data, we conclude that the intra/inter features as defined in (2) and (3) are good ones to characterize the spatial and the temporal correlations for inter/intra mode prediction. Also, the motion vector length plays an important role in the decision making process.

IV. Feature Space Partitioning

The 3-D feature space is partitioned into three regions (i.e., risk-free, risk-tolerable, and risk-intolerable regions) depending on the expected RD loss as

$$F = [f_0, f_1, |MV|] \in \begin{cases} R_{\text{free}} & L_p \le L_{\text{free}} \\ R_{\text{tolerable}} & L_{\text{free}} \le L_p \le L_p^* \\ R_{\text{intolerable}} & L_p^* \le L_p \end{cases}$$

where F denotes an input feature vector, L_p is the expected RD loss at position F, L_{free} is the threshold between the risk-free and risk-tolerable regions, and $L_p^* \in [0,1]$ is the threshold between the risk-tolerable and risk-intolerable regions. The expected RD loss is defined to be the normalized Lagrangian RD

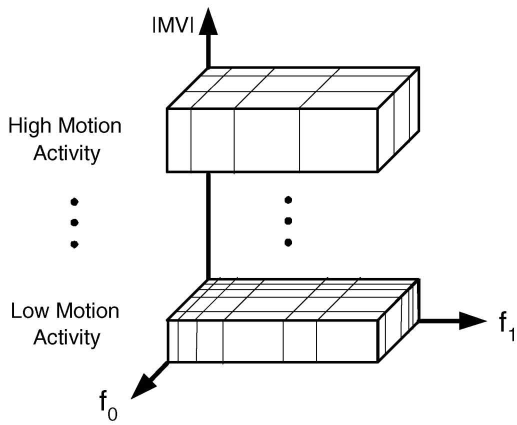


Fig. 5. Illustration of the partition of the 3-D feature vector space.

cost difference between the best mode and the wrongly selected mode

$$L_P = \frac{\sum (R_T - \hat{R}) + \lambda \cdot (D_T - \hat{D})}{\sum R_T + \lambda \cdot D_T}. \tag{8}$$

To facilitate the classification of an input feature vector to one of the three classes, it is convenient to partition the 3-D feature space based on a off-line training process, that is, we collect the three features as described in Section III from all macroblocks of seven training sequences of different motion and texture characteristics. They are: Akiyo, Hall Monitor, Foreman, Coastguard, Stefan, Table Tennis, and Mobile. The three features of a macroblock correspond to a point in the 3-D feature space.

There are two important factors to consider when we partition the feature space: representation accuracy and search complexity. For an efficient partition of the feature vector space, it is desirable to prevent empty cells which have no training data since it is difficult to calculate the expected RD loss in empty cells. The cell that does not have enough training data is also unfavorable since the expected RD loss may not be reliable. To avoid cells with little or no training data, we can employ vector quantization (VQ) for space partitioning. However, the complexity of VQ clustering is still high [25]. Here, we propose a nonuniform quantization scheme where each cell has about the same number of training data so that a reliable estimate for every cell with reasonable complexity can be obtained. This is illustrated in Fig. 5.

For given quantization parameter QP, the motion vector length is first nonuniformly quantized into N classes such that each motion class has about an equal number of training data with respect to the marginal probability. For each motion class i, the remaining two features (i.e., intra and inter features) are jointly quantized into $m_0^i \times m_1^i$ nonuniform cells, where $m_\tau^i = \lfloor \sqrt{|class_i|/M_\tau} \rfloor$ as shown in Fig. 5 with the product VQ technique. That is, cells are obtained by the tensor product of two independent 1-D partitions. Note that the number of cells can be different in different motion classes. The larger the cardinality of the ith class $|class_i|$, it is quantized into smaller cells. The training data per cell (M_τ) and the number of motion class (N) are chosen to minimize the RD performance degradation caused by wrong decision. The details are given in Section V-B.

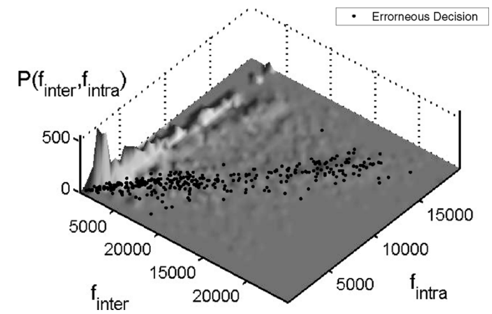


Fig. 6. Joint PDF of intra/inter features for correct decisions, including both intra or inter modes, and black dots represent scattered distribution of erroneous decision results along the diagonal boundary with the feature difference $\Delta f = 0$.

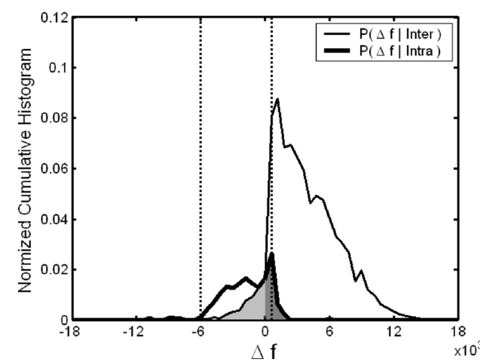


Fig. 7. Risk and risk-free region partitioning as a function of feature difference Δf.

V. Coding Mode Prediction

A. Risk-Free Region

As stated in Section IV, the 3-D feature space is partitioned into multiple classes according to the motion vector length. For a given motion class, we plot the distribution of the remaining two features, i.e., f_0 and f_1. A typical example is given in Fig. 6. The correct and erroneous decisions based on simple feature difference are labeled by shaded surface plot and solid dots, respectively.

We see that erroneous decisions primarily occur along the diagonal region, which is the boundary of these two features. It is also apparent that a large amount of macroblocks can be predicted without any RD loss using the feature difference as given in (5). We call the region that has a low probability of erroneous decision the risk-free region. For this region, the decision can be made simply based on the feature difference

$$\Delta f = f_{\text{Intra}} - f_{\text{Inter}} \underset{\text{Intra}}{\overset{\text{Inter}}{\gtrless}} 0. \tag{9}$$

To be more specific, the risk-free region is chosen under the criterion that the expected RD loss L_p is less than $L_{\text{free}} = 0.5\%$ in this study. This criterion is actually a conservative choice that guarantees no significant RD performance loss in the risk-free region.

Fig. 7 gives an example of risk and risk-free regions partitioned based on the expected RD loss. This subfigure shows two conditional PDFs of feature difference Δf when the inter or the intra mode is the best mode. The bold (or thin) curve at the left-hand (right-hand) side of $\Delta f = 0$ is the erroneous region of choosing the inter (intra) mode when the intra (inter) mode is the correct one. With $\Delta f = 0$ as the decision threshold, the erroneous decision regions are shaded by dark and light gray colors, respectively. The risk region corresponds to the interval inside the two dotted lines in Fig. 7, where the risk-free region consists of areas outside of the two dotted lines. We see that the risk region heavily overlaps with the shaded region of light gray. This is because the expected RD loss from the erroneously chosen intra mode is more severe than the erroneously chosen inter mode. This demonstrates that it is not sufficient to consider the conditional probabilities alone. We need to consider the cost of an erroneous decision as well.

B. Risk-Tolerable Region

To reduce the complexity associated with inter/intra mode decision, the risk region in Fig. 7 is further decomposed into two regions; namely, risk-tolerable and risk-intolerable regions, depending on the expected RD loss value. If the expected RD loss is less than a threshold, it is the risk-tolerable region. Otherwise, it is the risk-intolerable region. For the risk-tolerable region, we may develop an algorithm of medium complexity for mode decision. For the risk-intolerable region, the full mode search is performed to avoid significant RD loss. We will focus on the risk-tolerable case in this subsection.

In this study, we define the risk as

$$R = \sum_{i=0}^{M-1} \sum_{j=0}^{M-1} \tilde{C}_{ij} P(\hat{m}_i | m_j), \tag{10}$$

where m_j denotes the ground truth (or correct decision), $\hat{m}_i$ is the actual decision made, $\tilde{C}_{ij}$ is the cost of making decision $\hat{m}_i$, while the ground truth is m_j. We can rewrite (10) as

$$R = \sum_{i=0}^{M-1} \sum_{j=0}^{M-1} C_{ij} P(\hat{m}_i, m_j) \tag{11}$$

where $C_{ij} = \tilde{C}_{ij}/P(m_j)$. Furthermore, since decision $\hat{m}_i$ is made based on the feature space partition, we have

$$P(\hat{m}_i, m_j) = \int_{\chi_i} P(m_j | F) f(F) dF \tag{12}$$

where F denotes a vector in the feature space, χ_i is the subspace where decision $\hat{m}_i$ is chosen, and $f(F)$ is the probability density function of feature F. Substituting (12) into (11), we obtain

$$R = \sum_{i=0}^{M-1} \int_{\chi_i} \theta_i(F) f(F) dF \tag{13}$$

where

$$\theta_i(F) = \sum_{j=0}^{M-1} C_{ij} P(m_j | F) \tag{14}$$

which is called the Bayes risk since it is the sum of costs C_{ij} weighted by conditional probabilities $P(m_j|F)$ given an observed feature vector. In the current context, there are only two choices, and we use m_0 and m_1 to denote the decision of choosing the inter and the intra modes, respectively.

It is usually difficult to characterize the probability distribution $f(F)$ as given in (13). To simplify decision making, we simply focus on $\theta_i(F)$. By risk-minimizing mode selection, we mean the following mode-selection rule:

$$\theta_0(F) \underset{\text{Inter}}{\overset{\text{Intra}}{\gtrless}} \theta_1(F). \tag{15}$$

We have

$$\theta_0(F) = C_{01} P(m_1 | F) + C_{00} P(m_0 | F).$$

By setting $C_{00} = 0$, we obtain

$$\theta_0(F) = C_{01} P(m_1 | F) = C_{01} \frac{f(F|m_1) P(m_1)}{f(F)}$$

where the last equality is based on the Bayesian rule. Similarly, we have

$$\theta_1(F) = C_{10} P(m_0 | F) = C_{10} \frac{f(F|m_0) P(m_0)}{f(F)}.$$

Then, we can rewrite the decision rule in (15) as

$$\frac{f(F|m_1)}{f(F|m_0)} \underset{\text{Inter}}{\overset{\text{Intra}}{\gtrless}} \frac{C_{10} \cdot P(m_0)}{C_{01} \cdot P(m_1)}. \tag{16}$$

The left-hand side of (16) is the likelihood ratio while the right-hand side provides the decision threshold.

For each quantized cell in the risk-tolerable region, we determine the Bayes risk minimizing mode using the test in (16) with training data. After the risk-minimizing mode is determined, its expected RD loss is obtained using (8). In other words, we can compute the optimal mode and the associated expected RD loss pair (m_{opt}, L_P) for each cell using test sequences off-line and store these values in a lookup table (LUT).

VI. Probability Density Estimation

To determine the coding mode with (16), we need to estimate the likelihood ratio function that is the ratio of the posterior probabilistic density function. In general, there are three approaches to performing the likelihood estimation: the parametric, the semi-parametric, and the nonparametric approaches [26]. The parametric approach assumes a functional form of $f(F|m_i)$, e.g., the Gaussian mixture model, which is characterized by a set of parameters [27]–[30]. In this case, estimating $f(F|m_i)$ is essentially finding a set of parameters to fit the functional form to the training data. On the other hand, the nonparametric approach describes the observation data distribution directly. The semi-parametric approach, most notably used in neural networks, adopts a general functional form that has a variable number of adjustable parameters, e.g., the self-organizing map network [31]–[33]. In this study, we consider the parametric and the nonparametric likelihood estimation methods and compare them in terms of the RD performance and the computational complexity.

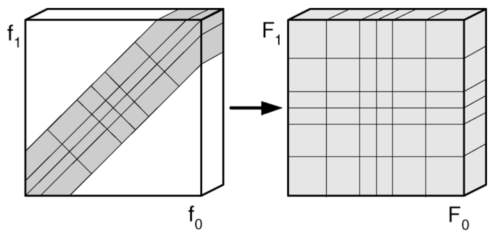


Fig. 8. Illustration of the remapping of the risk region.

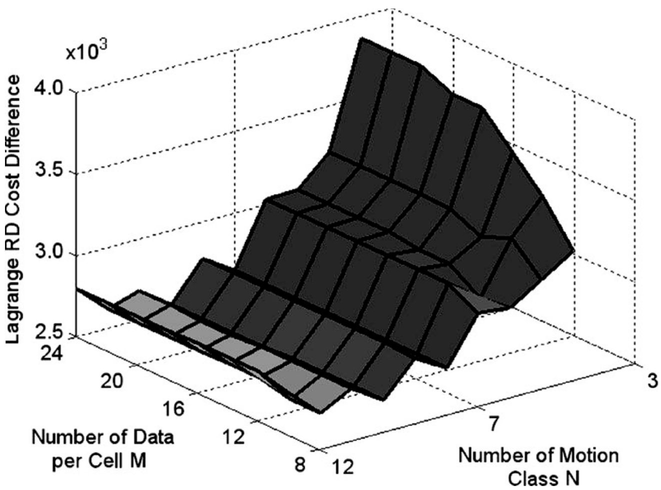


Fig. 9. Lagrange RD cost function plotted as a function of the number of motion class (N) and the average number of data per cell (M) for $QP = 28$.

A. Quantized Cells in 3-D Feature Space

As described in Section V-A, the two conditional distributions $f(F|m_i), i \in 0, 1$, overlap in the risk region as shown in Fig. 6. Thus, for a given motion class, it is desirable to remap the 2-D feature space $(f_{\text{Intra}}, f_{\text{Inter}})$ for compact quantization using the following coordinates transform:

$$\begin{bmatrix} F_0 \\ F_1 \end{bmatrix} = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} f_0 \\ f_1 \end{bmatrix}. \tag{17}$$

Fig. 8 illustrates the remapping from the original 2-D feature space of the risk region using (17).

As shown in Fig. 5, the remapped risk region is quantized into $m_i \times m_i$ cells each motion class i. On the one hand, if the risk region is coarsely partitioned, there are sufficient training data per cell and the density estimate will be more reliable. On the other hand, if the risk region is finely partitioned, the prediction error in each cell will be smaller, and it will be easier to make a correct mode decision in a smaller region. Consequently, there is a tradeoff in choosing a proper quantization scheme. The number of motion class (N) and the average number of data per cell (M) should be chosen carefully to minimize the expected RD loss. The RD cost difference for various M and N values is plotted in Fig. 9 when $QP = 28$. As shown in the figure, the proposed algorithm works best when $N = 9$ and $M = 8$.

B. Nonparametric Likelihood Estimation

To calculate the likelihood ratio in (16), the conditional probability density functions $f(F|m_0)$ and $f(F|m_1)$ of each quantized cell in the risk region can be estimated using the following normalized histograms [34]:

$$f(F|m_0) \cong \frac{H_{\text{Inter}}(x_q^i)}{N_0^i} \tag{18}$$

$$f(F|m_1) \cong \frac{H_{\text{Intra}}(x_q^i)}{N_1^i} \tag{19}$$

where $H_{\text{Inter}}(x_q^i)$ and $H_{\text{Inter}}(x_q^i)$ are the numbers of feature points that have the inter-prediction and intra-prediction as the best predictive modes, and N_0^i and N_1^i are the total numbers of inter and intra modes in motion class i, respectively. These density estimates are stored in an LUT that can be used by the decision rule in (16).

Please note that the density estimates may have irregularities when the training data are not sufficient to well represent the distribution in a quantized cell in the feature space. This problem can be alleviated by applying a 2-D smoothing filter, also called the smoothing kernel, to the obtained density estimates. The kernel function also helps reduce the variance of estimation errors. The selection of proper kernel functions has been studied extensively [35], [36], and a simple averaging filter can be used in practice [34]. Here, we adopt the following weighted averaging filter to obtain a regularized estimate:

$$\hat{P}(F^i|m_i) = \frac{\sum_{l=-k}^{k} \sum_{m=-k}^{k} \frac{h_i(F_q^i)}{d_{l,m}}}{\sum_{l=-k}^{k} \sum_{m=-k}^{k} \frac{1}{d_{l,m}}}$$

where $d_{l,m} = |F_q^i(0,0) - F_q^i(l,m)|_2$ is the Euclidean distance between the codeword of the current cell and the codeword in $(2k+1)^2$ neighboring cells in a quantized feature space.

C. Parametric Likelihood Estimation

The Gaussian mixture model (GMM) provides a powerful tool for density estimation due to its capability to represent various distributions. For each motion class, the 2-D feature vector space in the risk region is modeled as a mixture of multiple Gaussian distributions, that is, the probability of a GMM can be written as

$$\begin{aligned} P(F) &= \sum_{i=1}^{M} \omega_i \cdot N_i(F; \mu_i, \Sigma_i) \\ N_i(F; \mu_i, \Sigma_i) &= \frac{1}{(2\pi)^{|d|/2} |\Sigma_i|^{1/2}} \cdot e^{((1/2)(F-\mu_i)^T \Sigma_i^{-1} (F-\mu_i))} \\ \sum_i \omega_i &= 1, \forall i : \omega_i \geq 0 \end{aligned} \tag{20}$$

where M is the number of Gaussian mixtures, d is the dimension of input vector space, $N_i(\mu_i, \Sigma_i)$ is the ith component Gaussian probability density function with mean μ_i and covariance matrix

Σ_i, and ω_i is the prior probability of the ith component Gaussian pdf, i.e., $N_i(F; \mu_i, \Sigma_i)$.

The training of a GMM, i.e., finding a model for given feature vectors, is generally accomplished using the expectation-maximization (EM) algorithm, which guarantees convergence to a local maximum. The EM algorithm can be simply stated as follows. At each iteration, the likelihood is first obtained by the expectation of the complete-data likelihood with respect to the missing data using the current parameter estimates (E-step). Then, new parameter estimates are obtained by maximizing the marginal likelihood (M-step). The iterative EM process is performed until the model parameters converge. It is well known that the EM algorithm can be trapped to a local optimum and slow in convergence. To overcome these difficulties, we apply the K-means algorithm to find an initial distribution in the training process. Also, we used the full-rank GMM, where the covariance matrix of each Gaussian component has a full rank.

In this study, we train the GMM using the EM algorithm for each quantized cell in the risk region with multiple QP values, including 10, 16, 22, 28, and 34. Each conditional probability $f(F|m_i)$ is modeled as a GMM, and the number of component Gaussian pdfs is chosen to be the minimum number that achieves the minimum error with 50 iterations, when the error Δe is defined as the difference between old and new log-likelihoods. Each 2-D Gaussian pdf has a mean vector μ_i of dimension 2×1, a covariance matrix Σ_i of dimension 2×2, and a prior weight ω_i, so that the number of parameters to be estimated is seven per Gaussian component.

Those parameters are determined using the off-line EM algorithm and stored in an LUT. An example of the trained Gaussian mixture model for under the condition of $m = 0$ and $m = 1$ in the risk region is shown in Fig. 10, where the $QP = 22$ and the motion class is seven (out of a total of nine motion classes). The proposed intra/inter coding mode prediction algorithm can be summarized as shown in Table I.

Basically, once the coding mode is determined by the the proposed method, all feature values or intermediate values calculated for individual 4×4 blocks can be stored and reused for fine intra or inter mode decision if they are needed for further complexity reduction.

VII. Experimental Results

In the experiment, the proposed algorithm was integrated with the JVT reference software JM7.3a. For the experimental setup, the general main profile encoding configuration was used and the motion vector search range was set to 32×32 centered at the best predictive motion vector obtained by a fast full-search algorithm applied to five reference frames. The impact of parameter L_p^* on the RD performance, computational complexity and encoding time for the QCIF Table tennis sequence is shown in Figs. 11 and 12. Similarly, its impact on the RD performance and the computational complexity for the QCIF Foreman sequence with frame skip set to five (5fps), zero (30fps), and one (15fps) for Figs. 13–15, respectively. Also, five reference frames were used throughout the experiments. The B-frame was not used in the experiment. This algorithm can be applied to any profile since it switches the intra/inter mode prior to the coding of macroblocks. For the entropy coder, we chose CABAC [37].

The quantization parameter (QP) set $QP = \{10, 16, 22, 28, 34\}$ was chosen to cover a representative portion of the entire QP range, which is from 0 to 51. The test sequences were chosen to be MPEG sequences of classes A (i.e., News, Container), B (i.e., Foreman and Carphone), and

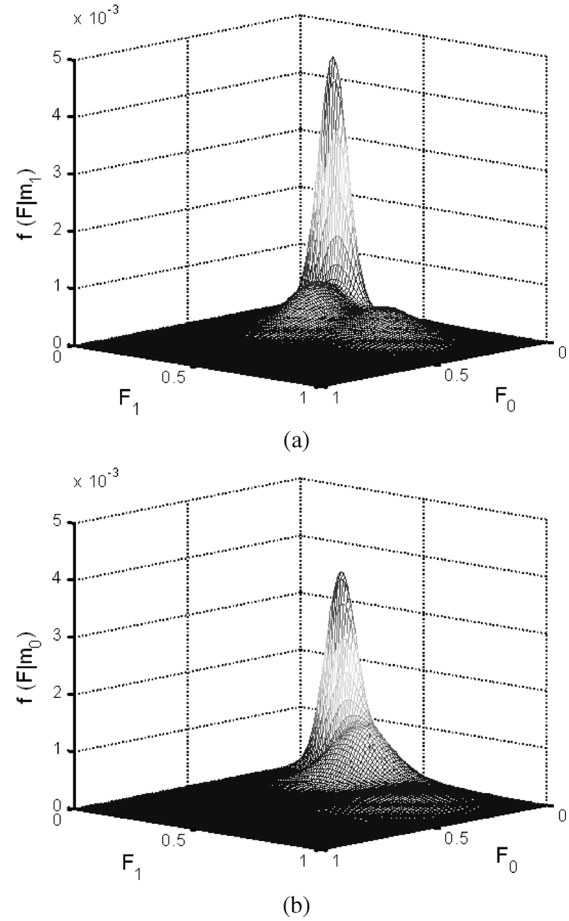


Fig. 10. Conditional probabilities of the trained GMM for motion class 7 with $\mathrm{QP} = 22$: (a) $f(F|m_0)$ and (b) $f(F|m_1)$.

TABLE I
BAYES-RISK MINIMIZED CODING MODE PREDICTION ALGORITHM

Step 1: For the current macroblock, calculate three features $[f_{Intra}, f_{Inter}, |MV|]$ using (2) and (3) and the Euclidean norm of the motion vector.
Step 2: Quantize the motion activity as described in sectionIV and check whether the feature vector is within the risk-free region. If yes, we will calculate the feature difference Δf and decide the mode based on (9). If no, we proceed to Step 3.
Step 3: Quantize the 2D feature vector in the risk region into small cells. If the non-parametric method is used, then determine the risk-minimized mode m_{opt} as desired mode, if associated RD loss $\bar{L}_P$ is less than L_p^* which means it belongs to risk-tolerable region. Otherwise, we proceed to Step 4. In case of parametric method, calculate likelihood ratio based on GMM trained by EM algorithm and determine risk minimized mode on the fly.
Step 4: Find the optimal intra mode and the optimal inter mode, respectively, using RDO or RD estimation algorithms. Then, the Lagrangian RD cost values for these two modes are compared and the one gives the smaller one is chosen to be the desired one.

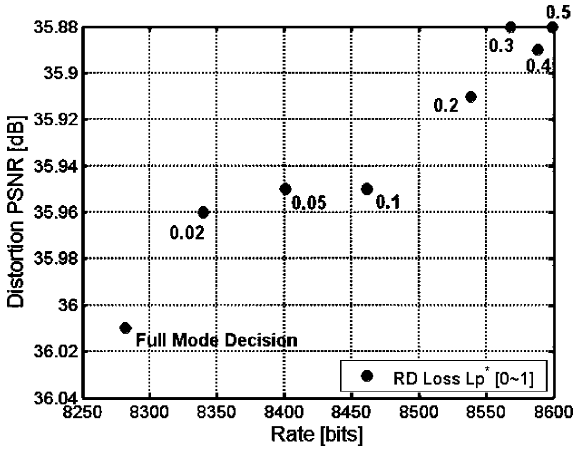
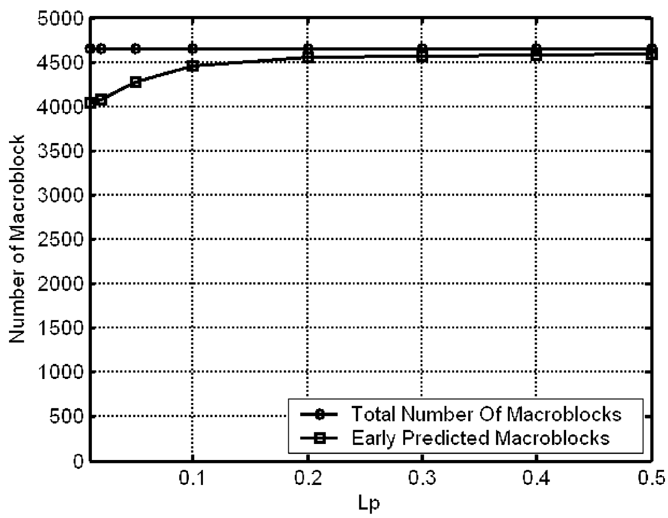
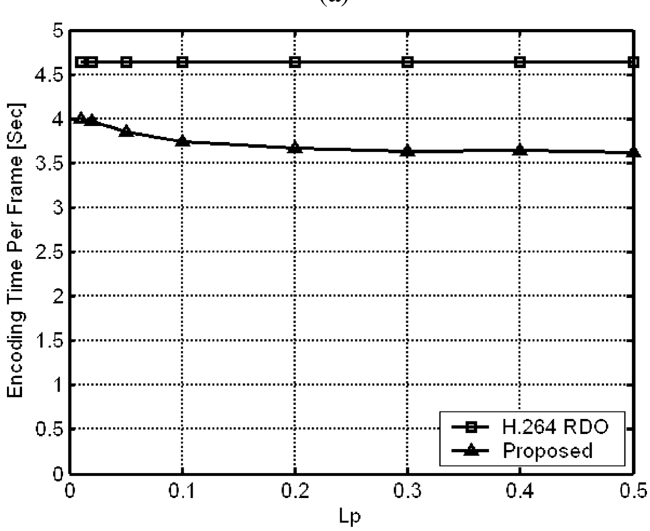


Fig. 11. RD performance as L_p^* increases for the QCIF Table Tennis sequence.

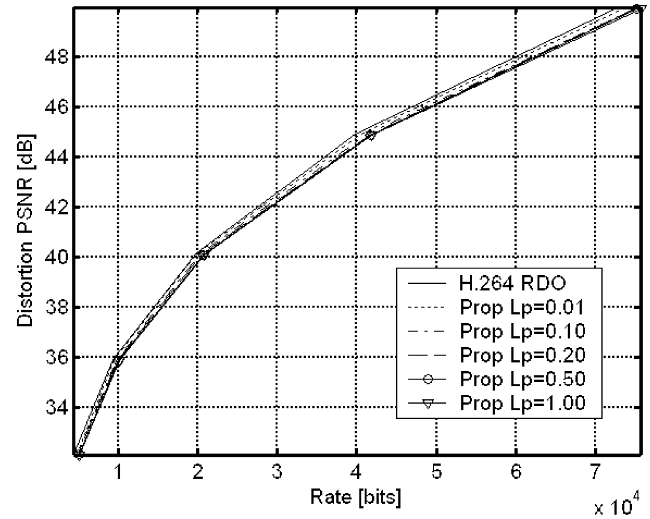


(a)

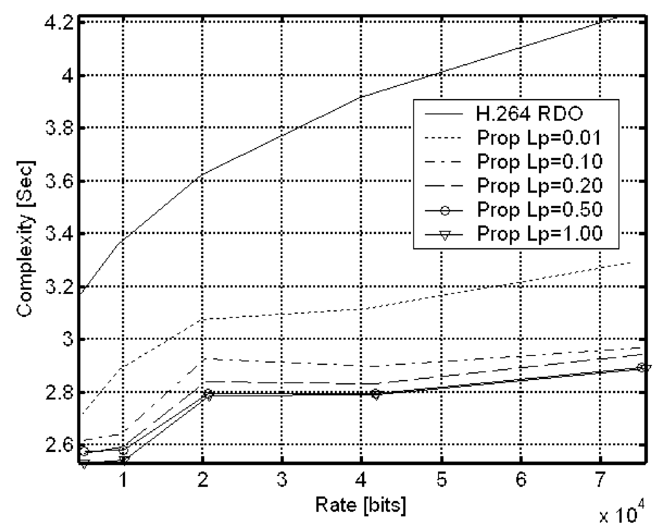


(b)

Fig. 12. (a) Computational complexity (square and triangular lines represent the number of single mode decisions). (b) Encoding time saving as L_p^* increases for the QCIF Table Tennis sequence.



(a)



(b)

Fig. 13. Performance comparison in (a) RD and (b) computational complexity as L_p^* increase for the QCIF Foreman sequence when frame-skip is set to 5.

C (i.e., Stefan and Football) sequence of various resolutions (from QCIF to D1) recommended in [38].

The simulation was conducted on a PC with Intel Pentium 4 Processor of speed 1.8 GHz, 512 MB DDR RAM. To compare the RD performance and the computational complexity of the proposed scheme with those of the RDO scheme in the H.264 reference code, the PSNR and the bit rate (per frame) were measured for a frame-skip set to five. When the frame-skip is small, the best mode is primarily the inter mode since the temporal correlation is more dominant than the spatial correlation. This is well reflected in the simple feature difference in (5), and the expected decision risk in the RD performance degradation is negligible.

In particular, when the frame-skip is 0 or 1 as shown in Figs. 14 and 15, it is confirmed by the experiment that most macroblocks fall in the risk-free region, and our algorithm quickly chooses the inter mode and the performance is excellent. When the frame-skip is large, the percentage of intra prediction is comparable with that of inter prediction since temporal correlation and spatial correlation are quite competitive in most macroblocks. Then, the risk of feature-difference-based

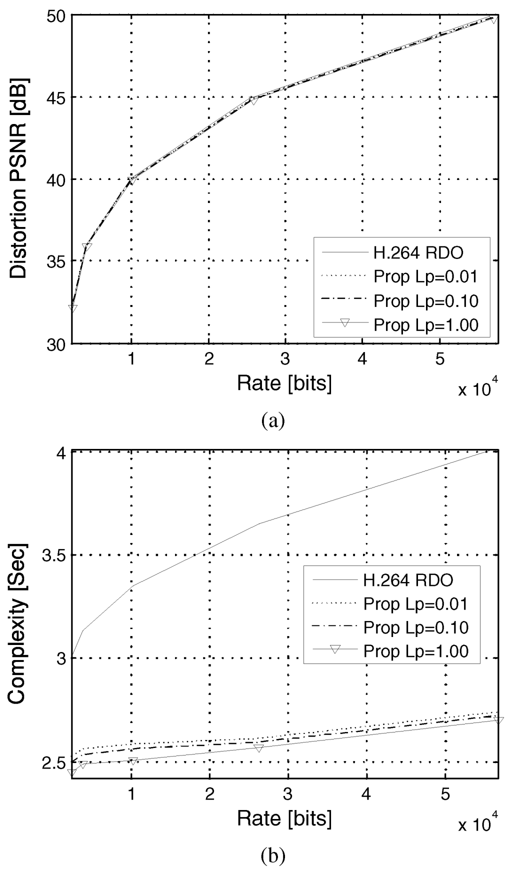


Fig. 14. Performance comparison in (a) RD and (b) computational complexity as L_p^* increase for the QCIF Foreman sequence when frame-skip is set to zero.

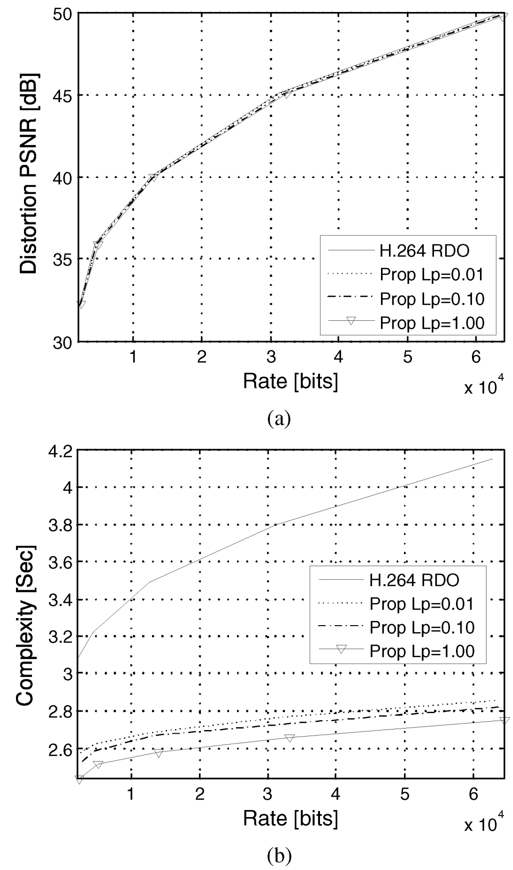


Fig. 15. Performance comparison in (a) RD and (b) computational complexity as L_p^* increases for the QCIF Foreman sequence when frame-skip is set to one.

decision is higher. The risk-minimizing decision scheme is developed to treat these difficult circumstances.

Hence, the reason to set the frame-skip to five is to evaluate the proposed method under the case where the inter and inter modes have more comparable performance. For computational complexity profiling, the encoding time was measured.The first experiment is to show the relationship between the rate-distortion-complexity (RDC) performance and the RD loss threshold L_p^*, which determines the boundary between the risk-tolerable and the risk-intolerable regions. As shown in Figs. 11 and 12, the RD performance loss increases and the computational complexity decreases when the RD loss threshold becomes larger, which allows a higher decision risk in the RD sense.

In other words, the risk-intolerable region is shrinking. It is worthwhile to mention that the contribution in terms of encoding time saving from the risk-free region ($L_p^* \simeq 0.01$) goes up to almost 60% of total encoding time saving at $L_p^* \simeq 0.5$, as shown in Fig. 12(b). This RDC performance trend is similarly observed in other sequences for a wide range of quantization parameters as shown in Fig. 13. We see that the RD performance loss increases slightly while the complexity decreases as we increase the tolerable RD loss threshold.

In Figs. 16 and 17, we compare the RDC performance tradeoff for three different algorithms, which are the RDO-based method, the proposed algorithm using the parametric as well as the nonparametric density estimation methods. As shown in these figures, the nonparametric method is more accurate and faster than the parametric method since the parametric method has to calculate the likelihood using GMM on the fly. On the other hand, in terms of the memory requirement, the nonparametric method needs more memory space since it needs to retrieve the risk-minimizing mode and the expected RD loss per cell from lookup table. For comparison, the parametric method only requires $7 \times N$ GMM parameters per cell, where N is the number of Gaussian mixture components and one Gaussian pdf needs seven parameters as described earlier.

The encoding bit rates, the PSNR values, and the speed-up factor for four test sequences of the proposed algorithm using the nonparametric and the parametric density estimation is compared with the RDO method in Tables II–IV. These table shows the result when RD loss threshold L_p^* is set as 1.0 and different RDC tradeoff can be achieved by adjusting one parameter which is RD loss threshold which partitions decision regions. L_p^*. For example, if we lower the L_p^* value, then we can achieve closer RD performance to RD optimized performance with less computational complexity savings and vice versa.

As shown in Tables II–IV, there are outliers that produce some loss, especially when QP is very large. However, considering the overall RD performance, we can achieve an average rate loss of 3.81% and an average distortion loss of -0.34% for all test sequences. Tables I and II show the result when the RD loss threshold Lp is set to 1.0. Please also note that the proposed

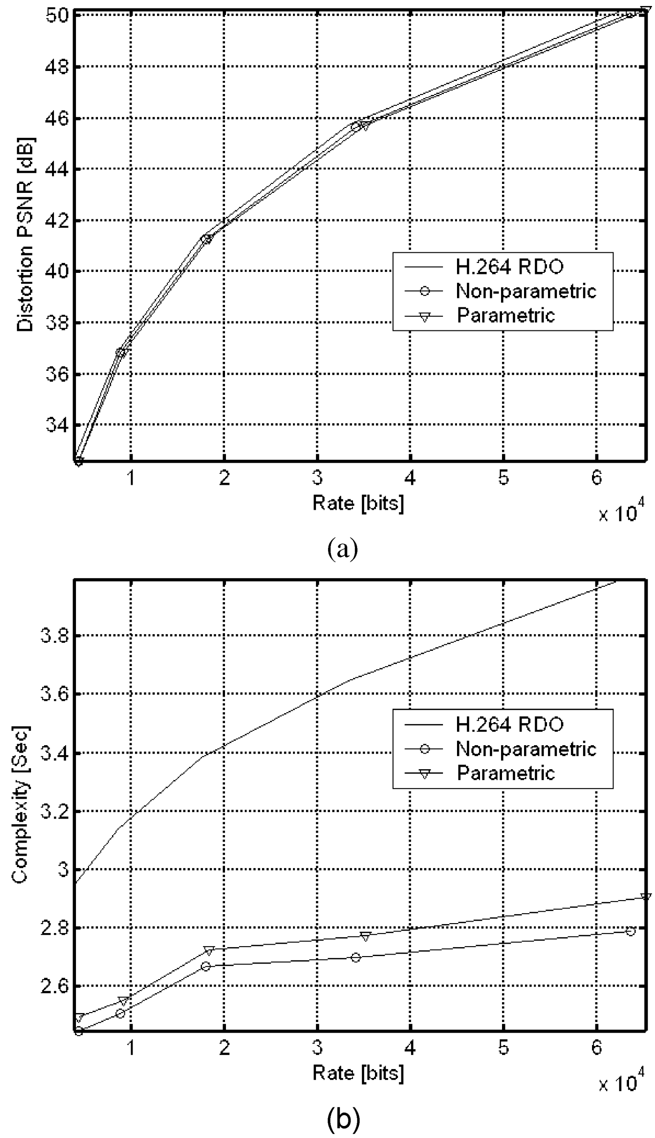


Fig. 16. Comparison of (a) the RD performance and (b) the computational complexity for the QCIF Carphone sequence.

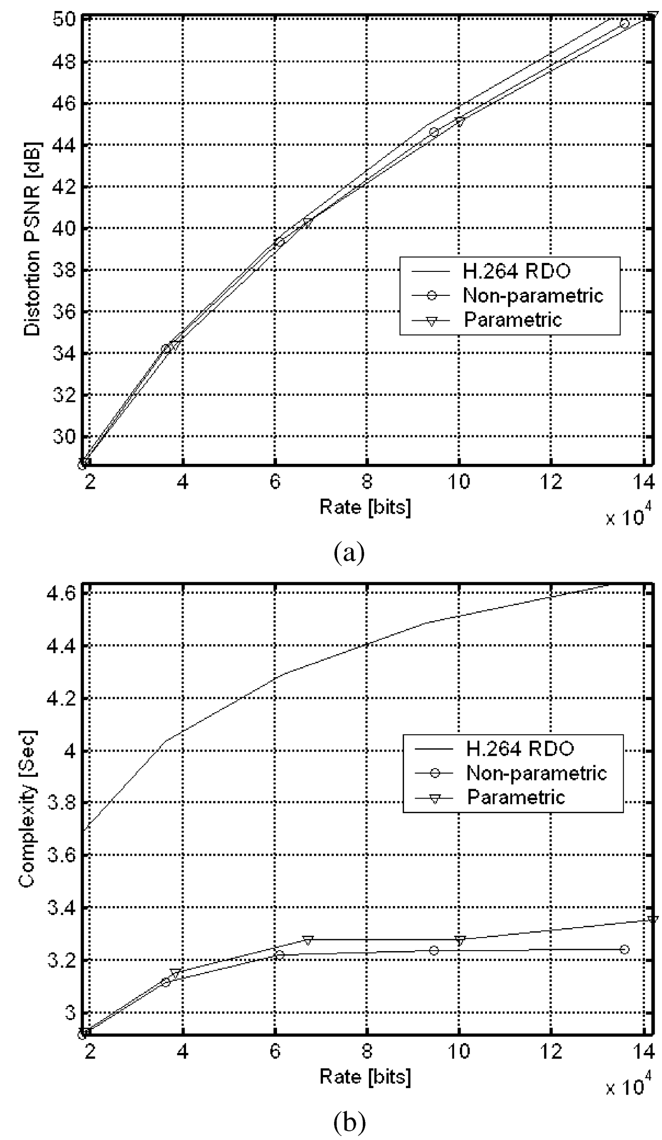


Fig. 17. Comparison of (a) the RD performance and (b) the computational complexity for the QCIF Stefan sequence.

algorithm allows the encoder to trade the computational complexity for video quality in a flexible way, i.e., by adjusting the Lp value, which determines the partition of risk regions. that is, by lowering the Lp value, we can achieve an RD performance closer to the RD optimized performance with less saving in the computational complexity.

VIII. Conclusion and Future Work

A feature-based intra/inter mode decision algorithm was proposed to speed up the H.264 encoding in this study. The main idea of the proposed coding mode prediction is to decide the mode using the expected risk of choosing the wrong mode in a multidimensional simple feature space. The proposed algorithm calculates three features and maps them into the one of three regions, namely, risk-free, risk-tolerable, and risk-intolerable regions. Depending on the mapped region, we can apply algorithms of different complexities for the final mode decision. Practically, the proposed algorithm selects the mode with

TABLE II
Rate Comparison (R: RDO; NP: Nonparametric; P: Parametric)

Test Sequences		Rate (Kbit/frame) [QP=10∼34]				
		10	16	22	28	34
News (QCIF)	R	35.89	20.45	11.35	5.83	2.94
	NP	36.66	21.15	11.83	6.09	3.21
	P	37.09	21.45	12.06	6.25	3.14
Foreman (QCIF)	R	72.70	39.82	19.83	9.33	4.51
	NP	75.65	41.83	20.80	10.12	5.12
	P	76.81	43.07	21.50	10.34	5.18
Carphone (QCIF)	R	62.49	33.27	17.53	8.53	3.93
	NP	63.65	34.09	17.99	8.79	4.28
	P	65.18	35.17	18.29	9.14	4.38
Stefan (QCIF)	R	133.0	92.78	61.63	36.29	18.15
	NP	135.9	94.52	61.08	36.39	18.16
	P	141.8	100.9	67.2	38.37	18.66
Container (CIF)	R	238.45	122.22	52.67	16.83	5.64
	NP	243.59	126.40	54.90	17.59	6.164
	P	249.1	130.5	58.76	18.04	6.044
Football (D1)	R	1638	1006	563.0	315.7	167.5
	NP	1698	1042	568.7	316.6	168.2
	P	1686	1045	574.4	336.3	175.5

TABLE III
DISTORTION COMPARISON (R: RDO; NP: NONPARAMETRIC; P: PARAMETRIC)

Test Sequences		PSNR (dB) [QP=10∼34]				
		10	16	22	28	34
News (QCIF)	R	50.17	45.86	41.34	36.78	32.38
	NP	50.14	45.83	41.31	36.77	32.31
	P	50.18	45.89	41.34	36.79	32.36
Foreman (QCIF)	R	49.98	44.94	40.19	35.93	32.17
	NP	49.92	44.86	40.10	35.89	32.13
	P	50.04	44.95	40.16	35.83	32.12
Carphone (QCIF)	R	50.19	45.71	41.36	36.90	32.66
	NP	50.11	45.62	41.25	36.84	32.61
	P	50.25	45.73	41.32	36.84	32.61
Stefan (QCIF)	R	50.05	44.94	39.72	34.33	28.87
	NP	49.80	44.61	39.34	34.23	28.66
	P	50.26	45.11	40.29	34.43	28.81
Container (CIF)	R	50.32	45.17	40.34	36.03	32.34
	NP	50.27	45.13	40.32	36.01	32.28
	P	50.32	45.18	40.34	36.02	32.31
Football (D1)	R	50.35	44.79	40.03	35.79	31.78
	NP	50.01	44.38	39.52	35.59	31.47
	P	50.46	44.88	40.52	35.81	31.63

TABLE IV
COMPLEXITY COMPARISON (R: RDO; NP: NONPARAMETRIC; P: PARAMETRIC)

Test Sequences		Speedup Factor (%) [QP=10∼34]				
		10	16	22	28	34
News (QCIF)	NP	32.26	29.63	26.06	23.77	19.59
	P	29.44	28.14	24.81	22.97	18.91
Foreman (QCIF)	NP	31.66	28.78	23.15	24.29	19.94
	P	25.71	26.95	21.18	23.47	19.32
Carphone (QCIF)	NP	30.13	25.95	21.03	20.14	16.86
	P	27.23	23.86	19.42	18.70	15.24
Stefan (QCIF)	NP	29.97	27.83	24.91	22.74	20.83
	P	27.57	26.88	23.58	21.74	20.50
Container (CIF)	NP	25.43	24.55	20.44	17.48	16.09
	P	23.90	22.96	19.04	16.86	15.13
Football (D1)	NP	25.08	22.45	20.46	17.85	17.27
	P	23.88	21.27	19.04	17.26	16.97

a lower risk in the RD sense using the Bayes-risk minimization criterion if the expected risk is less than a certain tolerance level. Otherwise, it performs the full mode decision to prevent significant RD performance loss. With the proposed algorithm, we demonstrated a speed-up factor of 20–32% for the JVT reference software JM7.3a without noticeable quality degradation. It is interesting to point out that the proposed algorithm allows the encoder to trade the computational complexity for video quality based on the characteristics of input video in a flexible way.

Our current work can be viewed a coarse-scale decision. If the proposed algorithm decides to go with the intra (or inter) prediction, the fast intra (or inter) mode prediction methods can be applied afterwards. In some sense, our algorithm and fast inter and intra mode prediction algorithms complement each other. The computational time reduction will not be as dramatic as presented if the proposed algorithm and fast mode prediction algorithms are used together. There are many fast mode decision algorithms proposed in the literature, and the interaction between these fast mode prediction algorithms and the proposed fast intra/inter coding mode selection can be very complex. It is an interesting future research topic to study the integration of the proposed intra/inter mode selection method with fast mode prediction algorithms for the overall performance improvement.

ACKNOWLEDGMENT

The authors would like to thank Dr. H.-H. Shih and Dr. F.-T. Chien for their helpful discussion to enhance the quality of this work.

REFERENCES

[1] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, “Overview of the H.264/AVC video coding standard,” *IEEE Trans. Circuits, Sys. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003.

[2] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer, and T. Wedi, “Video coding with H.264/AVC: Tools, performance, and complexity,” *IEEE Circuits Syst. Mag.*, vol. 4, no. 1, pp. 7–28, Jan. 2004.

[3] F. Pan, X. Lin, S. Rahardja, K. P. Lim, and Z. G. Li, “A directional field based fast intra mode decision algorithm for H.264 video coding,” in *Proc. IEEE Int. Conf. Multimedia Expo.*, Jun. 2004, vol. 2, pp. 1147–1150.

[4] Y.-W. Huang, B.-Y. Hsieh, T.-C. Chen, and L.-G. Chen, “Analysis, fast algorithm, and VLSI architecture design for H.264/AVC intra frame coder,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 3, pp. 378–401, Mar. 2005.

[5] A. C. Yu and G. R. Martin, “Advanced block size selection algorithm for inter frame coding in H.264/MPEG-4 AVC,” in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2004, vol. 1, pp. 95–98.

[6] Z. Zhou and M.-T. Sun, “Fast macroblock inter mode decision and motion estimation for H.264/MPEG-4 AVC,” in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2004, vol. 2, pp. 789–792.

[7] P. Yin, H.-Y. Tourapis, A. Tourapis, and J. Boyce, “Fast mode decision and motion estimation for JVT/H.264,” in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2003, vol. 3, pp. 853–856.

[8] Y.-K. Chen, A. Vetro, H. Sun, and S. Y. Kung, “Optimizing intra/inter coding mode decisions,” in *Proc. Int. Symp. Multimedia Inf. Process.*, 1997.

[9] D. S. Turaga and T. Chen, “Classification based mode decisions for video over networks,” *IEEE Trans. Multimedia*, vol. 3, no. 1, pp. 41–52, Mar. 2001.

[10] K.-P. Lim, G. Sullivan, and T. Wiegand, “Text description of joint model reference encoding methods and decoding concealment methods,” in *Proc. JVT 14th Meeting*, Hong Kong, China, 2005, ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6.

[11] A. Jagmohan and K. Ratakonda, “Time-efficient learning theoretic algorithm for H.264 mode selection,” in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2004, vol. 2, pp. 749–752.

[12] J. Lee and B. Jeon, “Fast mode decision for H.264,” in *Proc. IEEE Int. Conf. Multimedia Expo.*, Jun. 2004, vol. 2, pp. 1131–1134.

[13] P. Topiwala, G. Sullivan, A. Joch, and F. Kossentini, “Performance evaluation of H.26L TML 8 vs. H.263++ and MPEG4,” in *Proc. Video Coding Experts Group 15th Meeting*, Pattaya, Thailand, 2001, ITU-T Q.6/SG16 VCEG-042.

[14] M. Yang and W. Wang, “Fast macroblock mode selection based on motion content classification in H.264/AVC,” in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2004, vol. 2, pp. 24–27.

[15] Y. Shen, D. Zhang, C. Huang, and J. Li, “Fast mode selection based on texture analysis and local motion activity in H.264/JVT,” in *Proc. IEEE Int. Conf. Commun. Circuits Syst.*, Jun. 2004, vol. 1, pp. 27–29.

[16] C.-H. Kuo, M. Shen, and C.-C. J. Kuo, “Fast inter-prediction mode decision and motion search for H.264,” in *Proc. IEEE Int. Conf. Multimedia Expo.*, Jun. 2004, vol. 1, pp. 663–666.

[17] B. Meng and O. C. Au, “Fast intra-prediction mode selection for 4a blocks in H.264,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Apr. 2003, vol. 3, pp. 389–92.

[18] B. Meng, O. C. Au, C.-W. Wong, and H.-K. Lam, “Efficient intra-prediction mode selection for 4 × 4 blocks in H.264,” in *IEEE International Conference on Multimedia and Expo.*, Jul. 2003, vol. 3, pp. 521–4.

[19] Y. dong Zhang, F. Dai, and S. xun Lin, “Fast 4 × 4 intra-prediction mode selection for H.264,” in *Proc. IEEE Int. Conf. Multimedia Expo.*, Jun. 2004, vol. 2, pp. 27–30.

[20] C. Kim, H.-H. Shih, and C.-C. J. Kuo, “Fast H.264 intra-prediction mode selection using joint spatial and transform domain features,” *J. Vis. Commun. Image Representation*, vol. 17, pp. 291–310, 2006.

[21] C. Kim and C. C. J. Kuo, “Fast intra/inter mode decision for H.264 encoding using a risk-minimization criterion,” in *Proc. SPIE Applicat. Digital Image Process. XXVII*, Nov. 2004, vol. 5558, pp. 536–546.

[22] C. Kim and C.-C. J. Kuo, "A feature-based approach to fast H.264 intra/inter mode decision," in *IEEE Proc. Int. Symp. Circuit Syst.*, May 2005, vol. 1, pp. 308–311.

[23] C. Kim, H.-H. Shih, and C.-C. J. Kuo, "Feature-based intra-prediction mode decision for H.264," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2004, vol. 2, pp. 769–772.

[24] S. Zhu and K.-K. Ma, "A new diamond search algorithm for fast block-matching motion estimation," *IEEE Trans. Image Process.*, vol. 9, no. 2, pp. 287–290, Feb. 2000.

[25] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*, 159 ed. Boston, MA: Kluwer, 1992, Kluwer International Series in Engineering and Computer Science.

[26] C. M. Bishop, *Neural Networks for Pattern Recognition*. Oxford, U.K.: Clarendon, 1996.

[27] L. Xu and M. I. Jordan, "Unsupervised learning by EM algorithm based on finite mixture of Gaussians," in *Proc. World Congr. Neural Networks (II)*, 1993, pp. 431–434.

[28] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete date via the EM algorithm," *J. Roy. Statist. Soc. B*, vol. 39, pp. 1–38, 1977.

[29] R. A. Redener and H. F. Walker, "Mixture densities, maximum likelihood and the EM algorithm," *SIAM Rev.*, vol. 26, no. 2, pp. 195–239, 1984.

[30] D. Ormoneit and V. Tresp, "Averaging, maximum penalized likelihood and Bayesian estimation for improving Gaussian mixture probability density estimates," *IEEE Trans. Neural Netw.*, vol. 9, pp. 639–650, 1998.

[31] H. Yin and N. M. Allinson, "Comparison of a Bayesian SOM with the EM algorithm for Gaussian mixtures," in *Proc. Workshop Self-Organizing Maps*, 1997, pp. 304–305.

[32] ——, "Bayesian learning for self-organizing maps," *Electron. Lett.*, vol. 33, pp. 304–305, 1997.

[33] A. Utsugi, "Hyperparameter selection for self-organizing maps," *Neural Comput.*, vol. 9, pp. 637–648, 1997.

[34] D. Chai, S. L. Phung, and A. Bouzerdoum, "A Bayesian skin/non-skin color classifier using non-parametric density estimation," in *IEEE Proc. Int. Symp. Circuits Syst.*, May 2003, vol. 2.

[35] R. M. Gray and R. A. Olshen, "Vector quantization and density estimation," in *Proc. Compression Complexity Sequences*, Jun. 1997, pp. 172–193.

[36] Q. Xie, C. A. Laszlo, and R. K. Ward, "Vector quantization technique for nonparametric classifier design," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 12, pp. 1326–1330, Dec. 1993.

[37] D. Marpe, H. Schwarz, and T. Wiegand, "Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 620–636, Jul. 2003.

[38] "Evaluation Sheet for Motion Estimation," Draft version 4 JVT Test Model Ad Hoc Group, 2003, ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6.

Changsung Kim (S'96–M'06) received the B.S. and M.S. degrees from the Sung-Kyun-Kwan University, Korea, in 1995 and 1997, respectively, and the Ph.D. degree from the University of Southern California, Los Angeles, in 2006, respectively, all in electrical engineering.

He was an intern with Mitsubishi Electric Research Laboratories during the summer of 2005. He is presently a Senior Engineer with QUALCOMM MediaFLO Technologies, San Diego, CA. His research interests are in the areas of real-time video coding for mobile TV broadcasting technology and postprocessing techniques to improve subjective quality of digital video over wired or wireless network. He is the coauthor of four journal papers and 12 conference papers.

Dr. Kim is a member of the Society of Motion Picture Television Engineers. He served as technical reviewer for various IEEE conference and journal papers in 2003–2006.

C.-C. Jay Kuo (S'83–M'86–SM'92–F'99) received the B.S. degree in electrical engineering from the National Taiwan University, Taipei, Taiwan, R.O.C., in 1980 and the M.S. and Ph.D. degrees in electrical engineering from the Massachusetts Institute of Technology, Cambridge, in 1985 and 1987, respectively.

He is the Director of the Signal and Image Processing Institute (SIPI) and Professor of Electrical Engineering, Computer Science, and Mathematics, University of Southern California (USC), Los Angeles. His research interests are in the areas of digital image/video analysis and modeling, multimedia data compression, communication and networking, and biological signal/image processing. He is the coauthor of approximately 120 journal papers, 680 conference papers, and seven books.

Dr. Kuo is a Fellow of SPIE and a member of the Association of Computing Machinery. He is Editor-in-Chief of the *Journal of Visual Communication and Image Representation*and Editor of the *Journal of Information Science and Engineering*, *LNCS Transactions on Data Hiding and Multimedia Security*, and the *EURASIP Journal of Applied Signal Processing*. He was on the Editorial Board of the *IEEE Signal Processing Magazine* in 2003–2004. He served as Associate Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING from 1995 to 1998, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY from 1995 to 1997, and the IEEE TRANSACTIONS ON SPEECH AND AUDIO PROCESSING from 2001 to 2003. He was the recipient of the National Science Foundation Young Investigator Award (NYI) and the Presidential Faculty Fellow (PFF) Award in 1992 and 1993, respectively. He was the recipient of the Northrop Junior Faculty Research Award from the USC Viterbi School of Engineering in 1994. He received the Best Paper Award from the Multimedia Communication Technical Committee of the IEEE Communication Society in 2005. He was an IEEE Signal Processing Society Distinguished Lecturer in 2006. He is also Advisor to the Society of Motion Picture Television Engineers-USC Student Chapter.