Hindawi Publishing Corporation
EURASIP Journal on Applied Signal Processing
Volume 2006, Article ID 37175, Pages 1–10
DOI 10.1155/ASP/2006/37175

Scalable Fast Rate-Distortion Optimization for H.264/AVC

Feng Pan,[1, 2] Hongtao Yu,[3] and Zhiping Lin[3]

[1] *Media Processing Department, Institute for Infocomm Research, 21 Heng Mui Keng Terrace, Singapore 119613*
[2] *ViXS Systems Inc., 245 Consumers Road, Toronto, ON, Canada M2J 1R3*
[3] *School of Electrical & Electronic Engineering, Nanyang Technological University, 50 Nanyang Avenue, Singapore 639798*

Received 6 August 2005; Revised 17 March 2006; Accepted 27 May 2006

The latest H.264/AVC video coding standard aims at significantly improving compression performance compared to all existing video coding standards. In order to achieve this, variable block-size inter- and intra-coding, with block sizes as large as 16 × 16 and as small as 4 × 4, is used to enable very precise depiction of motion and texture details. The Lagrangian rate-distortion optimization (RDO) can be employed to select the best coding mode. However, exhaustively searching through all coding modes is computationally expensive. This paper proposes a scalable fast RDO algorithm to effectively choose the best coding mode without exhaustively searching through all the coding modes. The statistical properties of MBs are analyzed to determine the order of coding modes in the mode decision priority queue such that the most probable mode will be checked first, followed by the second most probable mode, and so forth. The process will be terminated as soon as the computed rate-distortion (RD) cost is below a threshold which is content adaptive and is also dependent on the RD cost of the previous MBs. By adjusting the threshold we can choose a good tradeoff between timesaving and peak signal-to-noise (PSNR) ratio. Experimental results show that the proposed fast RDO algorithm can drastically reduce the encoding time up to 50% with negligible loss of coding efficiency.

1. INTRODUCTION

H.264/AVC [1] is the newest international video coding standard developed by the joint video team (JVT), which consists of experts from VCEG and MPEG. It has achieved a significant improvement in coding efficiency compared to all the existing standards [2–4]. As in other video coding standards, H.264/AVC employs hybrid block-based motion compensated predictive coding. One of the novel features of H.264/AVC video coding is the use of different MB coding modes such as SKIP, INTER16 × 16, INTER16 × 8, INTER8 × 16, INTER8 × 8, INTRA16 × 16, and INTRA4 × 4, so that the temporal and spatial details in an MB are best presented. Note that in INTER8 × 8 mode, each block can be further divided independently into 8 × 8, 8 × 4, 4 × 8, or 4 × 4 subpartitions. To select the best coding mode, RDO is employed so that for each MB, all the MB coding modes are tried and the one that leads to the least RD cost is selected. This is to achieve the best tradeoff between the rate and distortion performance. Unfortunately, the computational burden of this type of exhaustively full searching algorithm is far more demanding than any other existing video coding standards.

As in the existing video coding standards, many efforts have been made in developing fast algorithms in motion estimation for H.264/AVC to reduce the complexity of H.264/AVC encoding [5–7]. Besides that, it is also possible to use fast mode decision strategy in H.264/AVC encoding. The basic idea of fast RDO-based mode decision in H.264/AVC is to select the coding mode that achieves the best RD performance without searching all the modes, leading to the reduction of computational complexity. This is based on the observation that a large MB partition suits slow motion and simple texture video objects, while a small partition size suits fast motion or complex scenes. Moreover, the occurrence of having different partition sizes in motion compensation is not equal, and can be decided by using the information of the temporal and spatial contents.

A number of efforts have been made to reduce the computational complexity of H.264/AVC by using various fast mode decision algorithms, such as fast SKIP mode decision [8], fast inter-mode decision [9–11], fast intra-mode decision [12–15], and the combination of the above [8, 16–19]. All the existing methods are based on the temporal correlation between current MB and its matching MB in the previous frame, and the spatial correlation between current MB and its neighboring MBs in the current frame. Therefore, these fast mode decision strategies are basically parallel such that the RD costs of all numbers or a reduced number of coding modes for an MB must be calculated before a decision can be made.

This paper has presented a new way for H.264/AVC fast mode decision. Unlike other methods which depend on temporal and spatial correlations, we study the probability distribution of coded modes. It is well known that, for most of the real-life video sequences, MB coding modes such as SKIP and INTER16 × 16 have much higher occurrences than the other coding modes. Thus in RDO process, we prioritize the MB coding modes such that the highest probable mode will be tried first, followed by the second highest probable mode, and so on. The MB coding mode with the least occurring probability will be tried at last. In this process, the computed RD cost will be checked against a content adaptive RD cost threshold to decide if we should terminate the RDO process before trying all of the possible modes. By adjusting this threshold we can actually control the time when the early termination can be activated, and thus this threshold can be used to determine the tradeoff between timesaving and PSNR loss. We could achieve a very significant timesaving by increasing this threshold with compromise of PSNR, or we can achieve a very good PSNR performance by reducing the threshold, but the timesaving will not be very significant. The advantage of the above fast RDO algorithm is that the order of MB coding mode is in accordance with the actual occurring probability of the MB coding modes. This can make sure that the more popular coding modes are checked prior to the less popular modes. In this way many unpopular coding modes are skipped, and the computational time is significantly reduced. Therefore, we can terminate the RDO process at any time when the RD cost is below a preset threshold. Another advantage of the proposed algorithm is that the threshold can be adjusted according to the user's preference of the tradeoff between timesaving and PSNR loss, and thus provides a scalable timesaving mechanism. Note that in our scheme, the RD costs of the MB coding modes are calculated and compared against a preset threshold one after another, until the early termination requirement is met. This differs fundamentally from other existing fast mode decision strategies where a reduced number of coding modes are tested and the one that produces the minimum RD cost is selected.

The rest of the paper is organized as follows. The next section describes the mode decision and RDO in H.264/AVC. Section 3 discusses the probability distribution of MB coding modes of test video sequences. Section 4 describes the proposed fast RDO algorithm based on the prioritization of MB coding modes and early termination. Section 5 shows the experimental results, and Section 6 is the conclusions.

2. OVERVIEW OF MODE DECISION IN H.264/AVC

In order to best represent the motion information and spatial details of an MB, H.264/AVC uses many different coding modes such as SKIP, INTER16 × 16, INTER16 × 8, INTER8 × 16, INTER8 × 8, INTRA16 × 16, and INTRA4 × 4. In INTER8 × 8 mode, each block can be further divided independently into 8 × 8, 8 × 4, 4 × 8, or 4 × 4 subpartitions. Figure 1 shows the different MB modes in H.264/AVC.

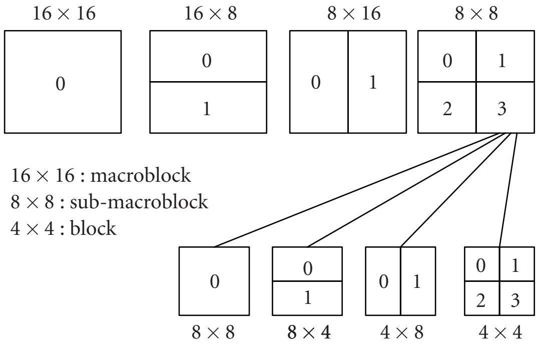


Figure 1: MB coding modes in H.264/AVC.

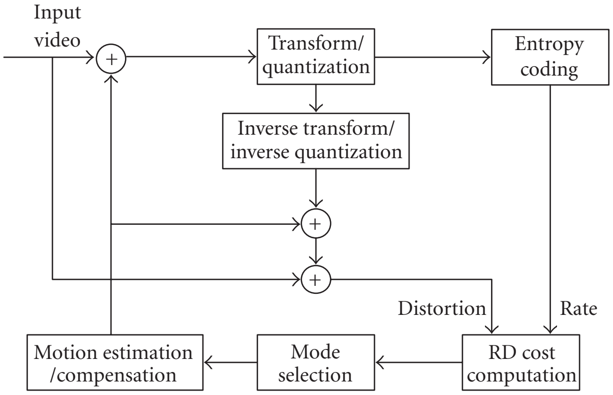


Figure 2: Calculation of RD cost using exhaustively full searching.

In the JVT reference model software, in order to choose the best MB coding modes, H.264/AVC makes use of full search RDO, which is very computationally expensive, and a Lagrangian multiplier method is used to achieve RDO. Figure 2 shows the procedure to achieve RDO using full search scheme. The detailed steps of this exhaustively full search RDO are as follows.

Step 1. Perform motion estimation for all the inter-modes.

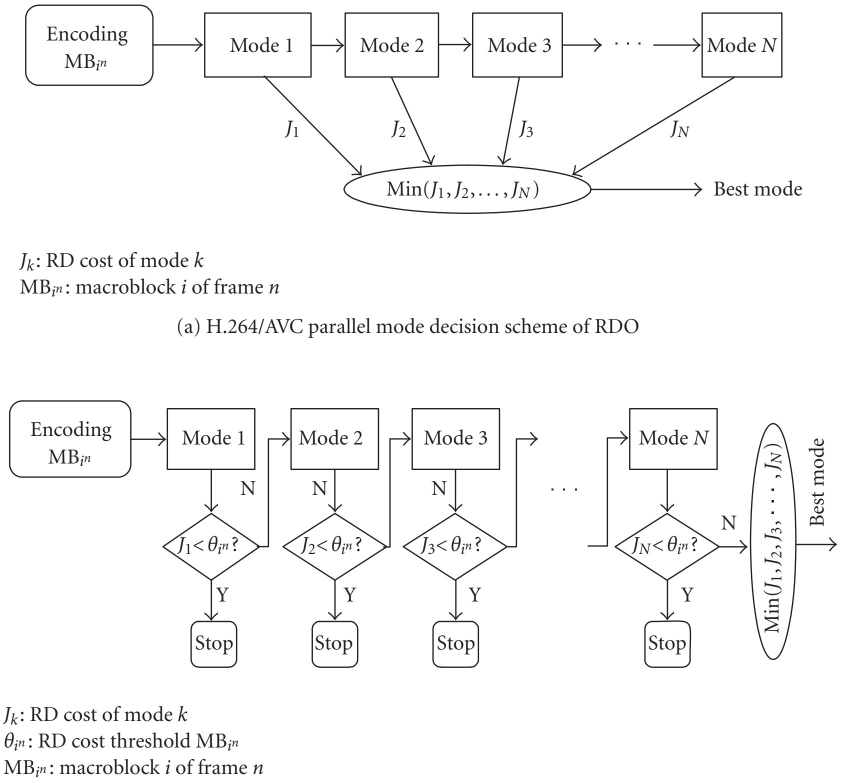
Step 2. Compute RD cost of all the coding modes. The RD cost of each mode J is calculated by using the number of bits R consumed by this MB and the sum of squared differences (SSD) between the original and the reconstructed pixels SSD:

$$J = \text{SSD} + \lambda \times R, \tag{1}$$

where λ is the Lagrangian multiplier.

Step 3. The coding mode that has the minimum J is selected as the best coding mode for this MB.

It can be seen from the above steps that the mode decision strategy of the full search RDO scheme is "parallel" such that RD costs of all coding modes for an MB must be


Encoding MB$_{i^n}$
Mode 1
Mode 2
Mode 3
· · ·
Mode N
J_1
J_2
J_3
J_N
Min($J_1, J_2, \ldots, J_N$)
Best mode
J_k: RD cost of mode k
MB$_{i^n}$: macroblock i of frame n
Encoding MB$_{i^n}$
Mode 1
Mode 2
Mode 3
Mode N
N
$J_1 < \theta_{i^n}$?
$J_2 < \theta_{i^n}$?
$J_3 < \theta_{i^n}$?
$J_N < \theta_{i^n}$?
Y
Stop
· · ·
Min($J_1, J_2, J_3, \cdots, J_N$)
Best mode
J_k: RD cost of mode k
θ_{i^n}: RD cost threshold MB$_{i^n}$
MB$_{i^n}$: macroblock i of frame n

(a) H.264/AVC parallel mode decision scheme of RDO

(b) Proposed sequential mode decision scheme of RDO

Figure 3: “Parallel” and “sequential” mode decision schemes.

calculated before a decision can be made. However, it is not necessary to test all the coding modes if the mode can be decided earlier by using the local content information of the video. For example, if the video object contain many detailed textures or high motion, the probability of coding it using small partition such as INTER8×8 mode is much higher than that of using a larger partition, and vice verse. In Section 4, we will present a more efficient mode decision scheme based on “sequential” decision strategy, such that we will compute the RD costs of MD coding modes one after another, in the order of descending occurring probability. This process will terminate immediately when the computed RD cost is below a threshold, which is adaptively decided by statistics of the neighboring MBs. Figure 3 shows the differences between the “parallel” and “sequential” mode decision strategies.

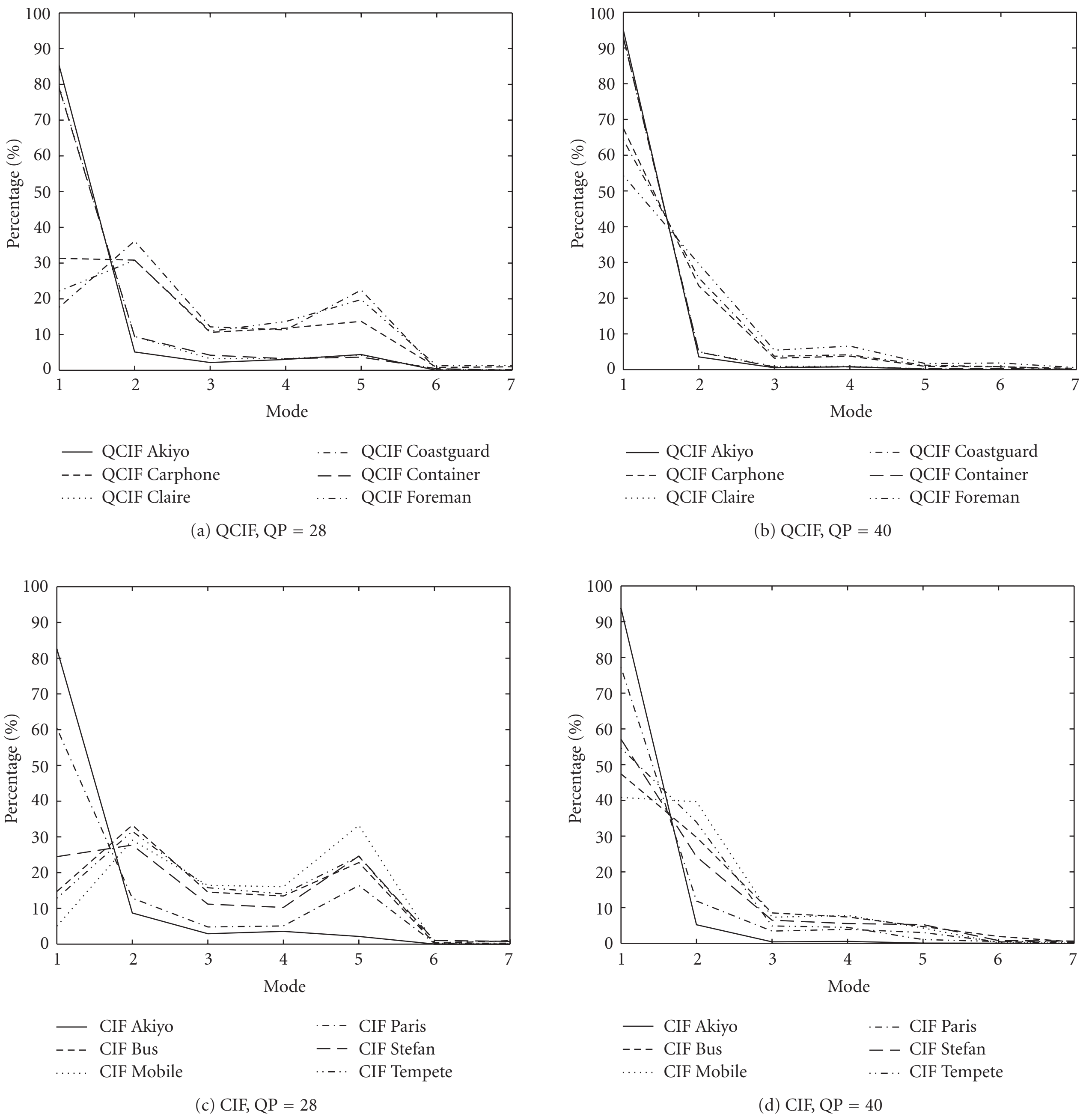
3. STATISTICAL STUDY ON MB CODING MODES IN H.264/AVC

If the best coding mode can be determined at an early stage of RD cost computation, significant timesaving can be achieved. The early termination strategy can be fulfilled making use of the local temporal and spatial contents, as well as a content adaptive threshold. In addition, motion estimation for any coding mode is performed only if there is a need to calculate the RD cost of this mode, and thus, the overall structure of the RDO process has to be modified to facilitate the early termination strategy.

3.1. Probability distribution of MB coding modes

It is observed that in encoding a natural video sequence, MBs in slow-motion and low-complexity frames are usually coded using larger partitions such as SKIP or 16 × 16, whereas MBs in fast-motion or high-complexity frames are likely to be coded using smaller partitions such as 8 × 8, 8 × 4, 4 × 8, or 4 × 4. Due to the strong temporal correlation between consecutive frames, the probability of encoding an MB using inter-mode is much higher than using intra-mode.

To verify the above observations, extensive experiments have been conducted on different sequences and at different quantization parameters (QP) to find out the statistics of MB coding modes in test video sequences. Figure 4 shows an example of the MB coding mode statistics of twelve test sequences by using full search RDO. In Figure 4, modes SKIP, INTER16 × 16, INTER16 × 8, INTER8 × 16, INTER8 × 8, INTRA16 × 16, and INTRA4 × 4 are represented by numbers 1 to 7, respectively. It can be seen from Figures 4(a) and 4(c) that for slow-motion sequences such as “Akiyo,” “Claire,” and “Container,” more than 85% of their MBs are encoded using the SKIP or INTER16 × 16 modes, and less than 5% of their MBs are encoded using INTER8 × 8, INTRA16 × 16,



(a) QCIF, QP = 28

(b) QCIF, QP = 40

(c) CIF, QP = 28

(d) CIF, QP = 40

Figure 4: Probability distribution of MB coding modes for different sequences.

and INTRA4 × 4 modes. On the other hand, for fast-motion and high-complexity sequences such as “Foreman,” “Mobile,” and “Stefan,” more than 40% of their MBs are encoded using the coding modes with smaller partitions. For example, the probabilities of using smaller partitions for “Foreman,” “Mobile,” and “Stefan” are 47%, 63%, and 47%, respectively. Furthermore, the probability of SKIP mode or the modes with larger partitions increases as QP increases, and the probability of the modes with smaller partitions decreases as QP decreases, which are shown in Figures 4(b) and 4(d). Therefore, significant timesaving can be achieved if we design an intelligent early termination strategy during RDO by taking into account the probability distribution of selected MB coding modes.

3.2. Mean value and standard deviation of RD cost

In RDO process, the RD cost of each MB coding mode must be computed in order to decide which mode would be eventually used. Thus an early termination strategy can be designed based on the RD cost of each coding mode. In order to activate the early termination correctly, we have conducted an experiment to explore the statistical properties of RD cost, such as their mean value and standard deviation,

Table 1: Mean value and standard deviation of RD cost for sequence “Foreman.”

Coding mode	QP = 28		QP = 32		QP = 36		QP = 40	
	Mean value	Standard deviation	Mean value	Standard deviation	Mean value	Standard deviation	Mean value	Standard deviation
SKIP	3684.53	2315.38	6825.61	4637.07	12696.27	9058.78	23508.09	17039.48
INTER16 × 16	5572.83	2854.27	10685.00	5534.81	20693.56	10551.22	39327.44	19721.68
INTER16 × 8	7000.06	2993.13	13683.81	5878.98	26175.07	10734.74	48750.11	19887.81
INTER8 × 16	6528.57	2810.95	12802.26	5554.29	24964.86	11003.29	48819.00	21537.71
INTER8 × 8	9432.36	3162.65	18392.22	5930.96	33771.91	11576.23	73151.46	23049.07
INTRA16 × 16	1234.20	962.37	3474.17	3199.58	930.49	7893.12	24144.37	17135.81
INTRA4 × 4	9756.36	3797.78	18599.62	6672.90	36940.29	10932.14	73401.14	19377.83

for different sequences under different quantization parameters (QP). Table 1 is the result for QCIF sequence “Foreman.” From this table, we can see that for different coding modes, the mean value and standard deviation of RD cost are quite different. In most cases, SKIP mode has the lowest mean value and standard deviation of RD cost, while INTRA4 × 4 mode has the highest mean value and standard deviation of RD cost. As QP increases, the mean value and standard deviation increase too. It can be seen that the standard deviation of RD cost has very large values and thus the RD cost of different MBs varies largely. Moreover, in most cases, the mean value of RD cost for most coding modes is in accordance with its occurring probability. The MB coding mode which has higher occurring probability usually produces lower mean value of RD cost. This shows that the mean value of RD cost is a good measure to distinguish different coding modes.

3.3. *Correlation coefficient of RD cost*

Although RD cost varies largely for different modes, the RD cost of neighboring MBs and their colocated MBs in the reference frame is highly correlated. This is evident from experiments. We use the correlation coefficient of RD cost between consecutive frames to represent the correlation of RD cost, which is defined as follows:

$$\rho = \frac{\mathrm{Cov}_{i,i-1}}{\sqrt{\mathrm{Var}_i \times \mathrm{Var}_{i-1}}}, \tag{2}$$

where $\mathrm{Cov}_{i,i-1}$ is the covariance of RD cost between Frame i's and Frame $i-1$'s MBs, Var_i is the variance of RD cost of Frame i's MBs, Var_{i-1} is the variance of RD cost of Frame $i-1$'s MBs. Note that Frame i and Frame $i-1$ are two consecutive frames. Figure 5 shows the correlation coefficient of RD cost between consecutive frames at QP = 28 of four different sequences. In Figure 5, the average values of correlation coefficient for all the sequences are larger than 0.9. The average correlation coefficient for less complicated sequences such as “Akiyo” is 0.983, and that for fast sequences such as “Stefan” is 0.952. This implies that the RD costs are very similar between consecutive frames, and thus provides a good basis for predicating the RD cost of the current frame's MB, which can be used to activate the early termination. That is, the statistical properties such as the mean value and standard deviation of RD cost of previous MBs can be used during the early termination in mode selection.

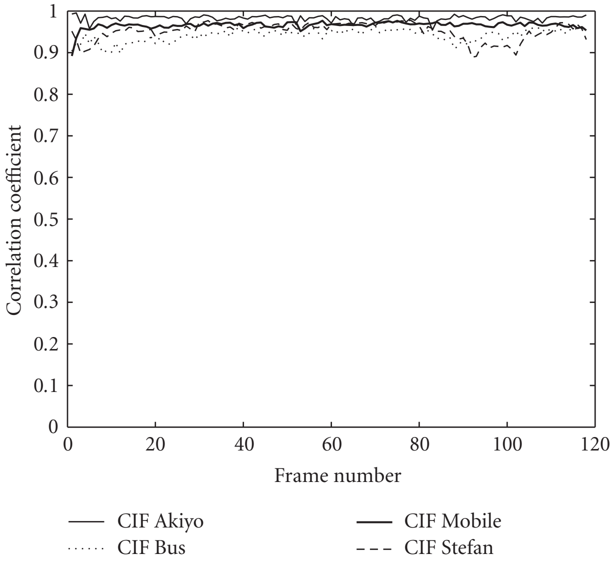


Figure 5: Correlation coefficient of RD cost.

4. PROPOSED PRIORITY-BASED FAST RDO ALGORITHM

4.1. *Prioritizing the MB coding modes*

Based on the observation as described in Section 3.1, we have designed an algorithm to sort the order of the MB coding modes for each MB to be coded according to their occurring probabilities. Occurring probability of a coding mode is the probability that the mode is selected as the best mode. Let n_i be the number of times that Mode i is selected as the best mode, let n be the total number of previously processed MBs, then the occurring probability of Mode i is

$$P_i = \frac{n_i}{n}. \tag{3}$$

The MB coding mode that has the highest occurring probability will be placed at the beginning of the queue, and will have the highest priority to be checked in mode decision process, while the MB coding mode that has the lowest occurring probability will be placed at the bottom of the queue and will be the last to be checked. If one coding mode has met the early termination criterion such that its RD cost is below the given threshold, this mode will be chosen as the best mode, and the remaining coding modes in the queue will be skipped. After that the priority queue will be updated, which will be used for mode decision of the next MB. Figure 6 illustrates this mechanism of prioritizing the MB coding modes.

Since the order of the coding modes in the priority queue is in accordance with the occurring probability of that mode, the more popular modes are always checked prior to the less popular modes. This ensures that the more popular modes are kept while the less popular modes are skipped if necessary. Thus the computational time is reduced.

4.2. *Early termination measure*

The objective of an early termination is to decide whether an MB coding mode has met the RD cost criterion so that the mode selection for the current MB can be terminated early without trying the rest of the coding modes in the queue. Based on the observations in Sections 3.2 and 3.3, an early termination mechanism according to the mean value and standard deviation of RD cost is defined as follows:

$$J \leq E_J - \alpha\sigma_J, \tag{4}$$

where J is the MB's RD cost as in (1), E_J is the mean value of the mode's RD cost, σ_J is the standard derivation of the same mode's RD cost, and α is a positive constant coefficient. Suppose the current MB is the nth MB, then

$$\begin{gathered} E_J = \frac{1}{n}\sum_{i=1}^{n} J_i, \\ \sigma_J = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(J_i - E_J\right)^2}. \end{gathered} \tag{5}$$

As we have shown, each coding mode has its own mean value of RD cost that is different from the one of the others, and the best coding mode has the minimum RD cost. If the RD cost of the current mode satisfies (4), this means that the current mode is, or very closed to, the optimal mode, and the PSNR loss will be negligible even if it is not the optimal mode. Therefore, the RDO process stops and the current mode is selected as the best coding mode. Here $E_J - \alpha\sigma_J$ means that the best coding mode approaches in the direction of reducing the average RD cost so that video quality is maintained. In (4), α is a parameter to control the video quality and computational time. If we want to save more time, α can be set to a lower value. On the other hand, if we want to maintain high video quality, α can be set to a higher value. Therefore, the adjusting of α makes our fast RDO algorithm scalable in terms of timesaving.

4.3. *Proposed fast RDO algorithm*

Based on the proposed prioritization mechanism and the early termination measure, the fast RDO algorithm is proposed as follows.

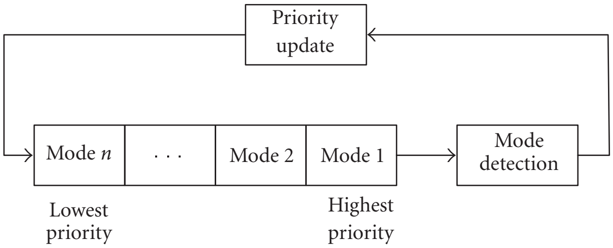


FIGURE 6: Priority mechanism.

Step 1. Sort the coding modes according to their occurring probability, and place them into the priority queue.

Step 2. Test the mode from the beginning of the priority queue. Compute its RD cost and check against early termination criterion.

Step 3. If the RD cost satisfies (4), select the current mode as the best mode (early termination) and go to Step 5.

Step 4. If the current mode is not the last mode in the priority queue, go to Step 2; otherwise, select the mode with the minimum RD cost as the best mode.

Step 5. Update the priority queue of the coding modes according to the new probability distribution.

Initially, for the first MB of the first P frame, all the modes are placed into the priority queue in the order of 1 to 7. Then a full search method is used to select the best coding mode for the first MB. After one coding mode has been selected as the best mode, the priority queue is updated according to the occurring probability of that mode such that the mode that has the highest occurring probability is placed at the beginning of the priority queue.

The mean value and standard deviation of RD cost are predicted dynamically according to that of the previous MBs. For the nth MB,

$$\begin{gathered} \hat{E}_{J,n} = \frac{1}{n}\left[(n-1)E_{J,n-1} + J_n\right], \\ \hat{\sigma}_{J,n} = \sqrt{\frac{1}{n}\left[(n-1)\sigma_{J,n-1}^2 + \left(J_n - \hat{E}_{J,n}\right)^2\right]}, \end{gathered} \tag{6}$$

where $E_{J,n-1}$ is the true mean value of RD cost for the previous $n-1$ MBs, $\sigma_{J,n-1}$ is the true standard deviation of RD cost for the previous $n-1$ MBs. Initially, $E_{J,0} = 0$, $\sigma_{J,0} = 0$.

One special case is the INTER8 × 8 mode. In checking the INTER8 × 8 mode, the 8 × 8 block will be further partitioned into smaller blocks such as 8 × 8, 8 × 4, 4 × 8, and 4 × 4. The RD cost of the subblocks is computed separately and their summation is the RD cost of mode INTER8 × 8. Therefore, no matter what the size of the subblock is, the coding mode is still considered as INTER8 × 8.

The proposed fast RDO algorithm is summarized in Figure 7.

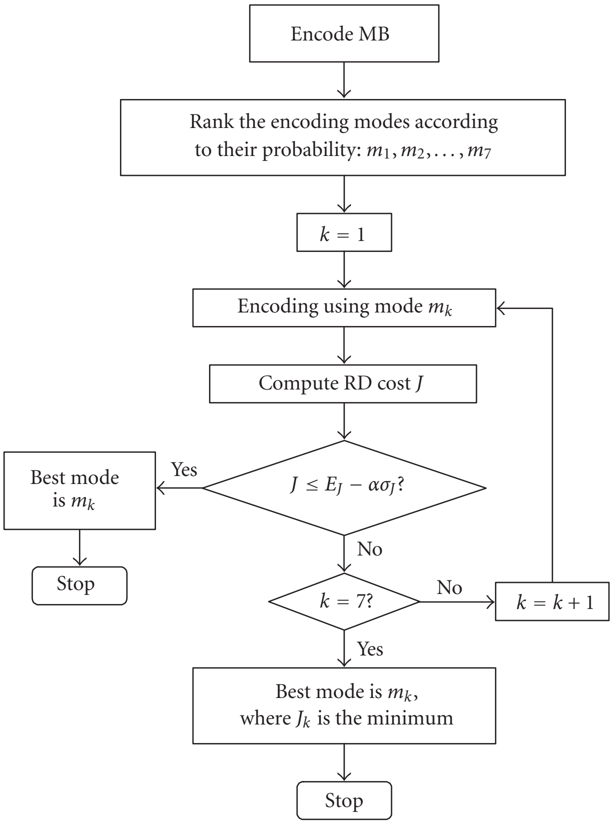

Encode MB
Rank the encoding modes according to their probability: $m_1, m_2, \ldots, m_7$
$k = 1$
Encoding using mode m_k
Compute RD cost J
$J \le E_J - \alpha\sigma_J$?
Yes
Best mode is m_k
Stop
No
$k = 7$?
No
$k = k + 1$
Yes
Best mode is m_k, where J_k is the minimum
Stop

FIGURE 7: Flowchart of the proposed fast RDO algorithm.

TABLE 2: Experimental conditions.

Frame type	IPPP...
Frame rate	30 fps
Slice mode	OFF
RDO	ON
Rate control	OFF
Hardmard	OFF
Search range	32
Restrict search range	No restriction
Symbol mode	CABAC
Partition mode	No data partition
Out File mode	Annex B

TABLE 3: Experimental results.

Sequence	Format	ΔPSNR (dB)	ΔBIT (%)	ΔTime (%)
Akiyo	QCIF	0.022	−0.362	−47.475
Carphone	QCIF	−0.052	1.086	−27.243
Claire	QCIF	−0.116	1.890	−49.062
Coastguard	QCIF	−0.090	2.551	−24.268
Container	QCIF	−0.116	2.300	−45.137
Foreman	QCIF	−0.139	2.472	−22.934
Akiyo	CIF	−0.004	0.118	−50.335
Bus	CIF	−0.109	2.281	−22.086
Mobile	CIF	−0.095	2.209	−20.604
Paris	CIF	−0.093	1.884	−32.327
Stefan	CIF	−0.099	2.037	−28.064
Tempete	CIF	−0.117	2.921	−23.327

5. EXPERIMENTAL RESULTS

To evaluate the performance of the proposed fast RDO algorithm, we compare it with JVT reference model software JM8.2 [20]. All the simulations are performed using a Pentium-43.00 GHz processor with 512 MB DDR RAM. The conditions of the experiment are listed in Table 2. In our experiments, we only consider the features available in the main profile of H.264/AVC.

In the experiments, the frame rate of the sequences is 30 frames per second. For QCIF sequences, the number of frames is 240. For CIF sequences, the number of frames is 120. For each sequence, four QP values of 28, 32, 36, and 40 are used. The comparison results are produced and tabulated based on the averaged difference of coding time (ΔTime), the averaged PSNR difference (ΔPSNR), and the averaged bit rate difference (ΔBIT). In order to evaluate the timesaving of the fast RDO algorithm, the time difference is defined as follows.

For QP_i ($i = 1, \ldots, 4$), let $T_{\text{JM},i}$ denote the coding time used by JM8.2 encoder and let $T_{\text{FR},i}$ be the time taken by the fast RDO algorithm, the difference of coding time is defined as

$$\Delta\,\text{Time}_{\,i} = \frac{T_{\text{FR},i} - T_{\text{JM},i}}{T_{\text{JM},i}} \times 100\%. \tag{7}$$

Thus the average difference of coding time is as follows:

$$\Delta\,\text{Time} = \frac{1}{4}\sum_{i=1}^{4} \Delta\,\text{Time}_{\,i} \tag{8}$$

PSNR and bit rate differences are calculated according to the numerical averages between the RD-curves derived from JM8.2 encoder and the fast RDO algorithm, respectively. The detailed procedures in calculating these differences can be found in [21], which is recommended by JVT Test Model Ad Hoc Group [22]. Note that PSNR and bit rate difference should be regarded as equivalent, that is, there is either the increase in PSNR or the decrease in bit rate, not both at the same time.

The experimental results with $\alpha = 0.3$ are given in Table 3. As can be seen from Table 3, our algorithm has achieved a significant saving in the average encoding time compared to JM8.2, while at the same time the loss of video quality is negligible.

Table 4 shows the detailed performance results for CIF sequence "Paris" for different QPs. As QP increases, the amount of timesaving increases. This is because in this case, the probability of SKIP mode increases. Since SKIP mode has the highest priority, it is checked prior to other coding modes. Thus the timesaving will be more significant than in the case of lower QPs.

Table 4: Performance results for sequence "Paris."

QP	PSNR difference (dB)	Bit rate difference (%)	Time difference (%)
28	−0.100	−0.074	−28.654
32	−0.100	−0.162	−30.466
36	−0.110	−0.230	−33.644
40	−0.080	−0.050	−36.545

(a) Original image

1	1	1	1	4	5	5	3	2	4	2
1	1	1	2	1	1	5	1	2	2	3
2	1	1	1	1	1	3	5	2	2	4
1	1	5	2	2	4	2	5	2	2	2
4	2	5	5	2	1	3	4	2	2	4
1	1	3	5	1	1	1	4	1	2	2
1	1	2	5	5	3	3	2	2	2	4
1	1	1	3	5	5	4	3	2	2	2
4	4	2	2	4	4	5	5	5	3	3

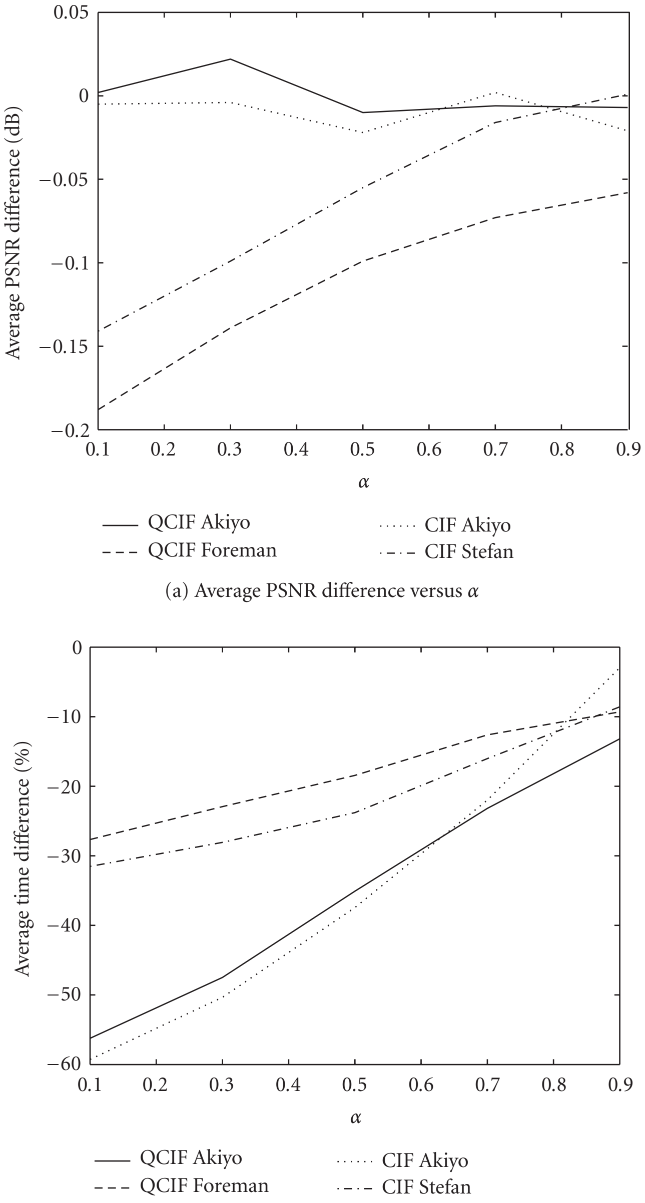
(b) MB coding modes by JM8.2

1	1	1	1	3	5	5	2	2	3	4
3	3	2	2	1	1	5	2	2	2	2
2	1	1	2	1	1	2	5	3	2	1
1	2	5	3	2	5	2	5	3	2	1
1	2	3	5	3	1	2	4	2	1	1
1	1	2	4	3	1	2	1	2	2	3
1	1	2	4	3	3	5	1	2	2	2
1	1	1	5	5	5	4	4	3	2	2
4	4	2	2	5	4	4	5	5	3	5

(c) MB coding modes by proposed algorithm

Figure 8: The mode distribution for the 45th frame of sequence "Foreman."

Figure 8 shows the coding modes of different MBs in the 45th frame of QCIF sequence "Foreman." Figure 8(b) is the result without using fast mode decision scheme; Figure 8(c) is the result with the proposed fast RDO algorithm. It can be seen from these figures that there are nearly 60% MBs that have exactly the same coding modes, and the others are having the similar coding modes. We define that the two coding modes are similar if their block sizes are next to each other. For example, 8 × 8 is similar to 8 × 16 and 16 × 8. Although not all the modes are the same, the PSNR loss of the proposed algorithm is only 0.042 dB. This high similarity in MB coding modes of these two schemes shows that the proposed fact RDO is effective.



(a) Average PSNR difference versus α

(b) Average time difference versus α

Figure 9: The average PSNR difference and average time difference.

Figures 9(a) and 9(b) give the results of average PSNR difference versus α and average time difference versus α, respectively. In Figure 9(a), for high motion sequences "Foreman" and "Stefan," the average PSNR difference increases as α increases. In Figure 9(b), for all the sequences, the average time difference decreases as α increases. This shows that α can be used as a control parameter for the tradeoff between the reconstructed video quality and computational complexity.

Table 5: Comparison between [19]'s algorithm and proposed algorithm.

Sequence	ΔPSNR (dB)		ΔBIT (%)		ΔTime (%)	
	[19]	Proposed	[19]	Proposed	[19]	Proposed
Akiyo	−0.25	−0.00	5.41	0.12	−48.82	−50.34
Bus	−0.19	−0.11	3.92	2.28	−20.28	−22.09
Mobile	−0.18	−0.10	3.86	2.21	−15.77	−20.60
Stefan	0.10	−0.10	1.96	2.04	−15.87	−28.06

If we want to save more time, we can decrease α. Otherwise, α can be increased to retain the coding quality.

In Table 5, we compare our proposed algorithm with that of Lu et al. [19]. In [19], Lu et al. proposed a fast mode decision algorithm for B and P frames in H.264, where the information from the previously coded MBs, such as neighboring mode, residue, and RD cost, is used to determine that some of the modes can be skipped in the RDO process. The choice of early termination thresholds depends on the fixed mode order that is determined before the coding. Note that in our algorithm, the order of modes is based on the mode popularity that is updated adaptively during the RDO process.

For sequence "Akiyo," "Bus," and "Mobile," our proposed algorithm performs better in PSNR, bit rate, and timesaving. As for sequence "Stefan," although [19]'s algorithm performs better in terms of PSNR, our proposed algorithm has much significant timesaving compared to that of [19].

6. CONCLUSIONS

In this paper, we have proposed a fast RDO algorithm based on the occurring probability of different coding modes. The coding mode which has higher occurring probability will be tried first in the RDO process. Once the RD cost of the coding mode has met the early termination criterion, the RDO process will be stopped immediately without testing the rest of coding modes in the priority queue, thus significant timesaving can be achieved. By adjusting a threshold which is proportional to the RD cost of the previously encoded frame, we can achieve good tradeoff between timesaving and PSNR loss, and thus this approach is scalable. Simulation results have shown that our proposed algorithm achieves significant timesaving with negligible PSNR loss when compared with JM8.2.

REFERENCES

[1] ISO/IEC JTC1, *Information Technology—Coding of Audio-Visual Objects—Part 10: Advanced Video Coding*, ISO/IEC FDIS 14496-10, 2003.

[2] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, 2003.

[3] A. Puri, X. M. Chen, and A. Luthra, "Video coding using the H.264/MPEG-4 AVC compression standard," *Signal Processing: Image Communication*, vol. 19, no. 9, pp. 793–849, 2004.

[4] G. J. Sullivan and T. Wiegand, "Video Compression—from concepts to the H.264/AVC standard," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 18–31, 2005.

[5] H. Y. C. Tourapis and A. M. Tourapis, "Fast motion estimation within the H.264 codec," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME '03)*, vol. 3, pp. 517–520, Baltimore, Md, USA, July 2003.

[6] C.-F. Lin and J.-J. Leou, "An adaptive fast full search motion estimation algorithm for H.264," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS '05)*, vol. 2, pp. 1493–1496, Kobe, Japan, May 2005.

[7] C.-W. Lam and L.-M. Po, "Fast block motion estimation with early acceptance technique in H.264/JVT," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS '05)*, vol. 2, pp. 1513–1516, Kobe, Japan, May 2005.

[8] B. W. Jeon and J. Y. Lee, "Fast mode decision for H.264," in *Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG 8th Meeting*, Waikoloa, Hawaii, USA, December 2003, Document JVT-J033.

[9] K. P. Lim, S. Wu, D. J. Wu, et al., "Fast inter mode selection," in *Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG 9th Meeting*, San Diego, Calif, USA, September 2003, Document JVT-I020.

[10] Z. Zhou and M.-T. Sun, "Fast macroblock inter mode decision and motion estimation for H.264/MPEG-4 AVC," in *Proceedings of the IEEE International Conference on Image Processing (ICIP '04)*, vol. 5, pp. 789–792, Singapore, Singapore, October 2004.

[11] D. Wu, F. Pan, K. P. Lim, et al., "Fast intermode decision in H.264/AVC video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 7, pp. 953–958, 2005.

[12] C.-H. Tseng, H.-M. Wang, and J.-F. Yang, "Improved and fast algorithms for intra 4 × 4 mode decision in H.264/AVC," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS '05)*, vol. 3, pp. 2128–2131, Kobe, Japan, May 2005.

[13] C.-C. Cheng and T.-S. Chang, "Fast three step intra prediction algorithm for 4×4 blocks in H.264," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS '05)*, vol. 2, pp. 1509–1512, Kobe, Japan, May 2005.

[14] F. Pan, X. Lin, S. Rahardja, et al., "Fast mode decision algorithm for intraprediction in H.264/AVC video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 7, pp. 813–822, 2005.

[15] Y. Su, J. Xin, A. Vetro, and H. Sun, "Efficient MPEG-2 to H.264/AVC intra transcoding in transform-domain," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS '05)*, vol. 2, pp. 1234–1237, Kobe, Japan, May 2005.

[16] Y. J. Liang and K. El-Maleh, "Low-complexity Intra/Inter mode-decision for H.264/AVC video coder," in *Proceedings of the International Symposium on Intelligent Multimedia, Video and Speech Processing (ISIMP '04)*, pp. 53–56, Hong Kong, China, October 2004.

[17] C. Kim and C.-C. J. Kuo, "A feature-based approach to fast H.264 intra/inter mode decision," in *Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS '05)*, vol. 1, pp. 308–311, Kobe, Japan, May 2005.

[18] E. Arsura, L. Del Vecchio, R. Lancini, and L. Nisti, "Fast macroblock intra and inter modes selection for H.264/AVC," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME '05)*, Amsterdam, The Netherlands, July 2005.

[19] X. A. Lu, A. M. Tourapis, P. Yin, and J. Boyce, "Fast mode decision and motion estimation for H.264 with a focus on MPEG-2/H.264 transcoding," in *Proceedings of the IEEE International*

Symposium on Circuits and Systems (ISCAS '05), vol. 2, pp. 1246–1249, Kobe, Japan, May 2005.

[20] JVT, Reference Model JM8.2, ftp://standards.polycom.com.

[21] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," in *Proceedings of the ITU-T VCEG 13th Meeting*, Austin, Tex, USA, April 2001, Document VCEG-M33.

[22] VT Test Model Ad Hoc Group, "Evaluation sheet for motion estimation," Draft version 4, February 2003.

Feng Pan received the B.S., M.S., and the Ph.D. degrees in communication and electronic engineering from Zhejiang University, China, in 1983, 1986, and 1989, respectively. Since then, he has been teaching and researching in a number of universities in China, UK, Ireland, and Singapore. He is currently working as a video architect in ViXS Systems Inc., Canada. His research areas are digital image processing, digital image/video compression, digital television broadcasting, and pattern recognition. He has published more than 70 technical papers, and conducted many short courses for industry in the above areas. He was the General Chairman of the 7th International Symposium on Consumer Electronics, Sydney, Australia, December 3–5, 2003, and has been serving as the Member of organizing committee for a number of international conferences. He was the Chapter Chairman of IEEE Consumer Electronics, Singapore, from 2002 to 2004. He is currently the Associate Editor of the International Journal of Innovational Computing and Information Control.

Hongtao Yu received his B.Eng. degree from Huazhong University of Science and Technology, and his M.Eng. degree from Huazhong University of Science and Technology and Nanyang Technological University, respectively. He is now with Nanyang Technological University. His research areas are digital image processing, digital image/video compression, multimedia system, and telecommunication network management.

Zhiping Lin received the Ph.D. degree in information engineering from the University of Cambridge, England, in 1987. He was with the University of Calgary, Canada, from 1987 to 1988, with Shantou University, China, from 1988 to 1993, and with DSO National Laboratories, Singapore, from 1993 to 1999. Since February 1999, he has been an Associate Professor at Nanyang Technological University (NTU), Singapore. He is also the Program Director of Bio-Signal Processing, Center for Signal Processing, NTU. He was an Editorial Board Member of Multidimensional Systems and Signal Processing from 1993 to 2004, and a Coeditor of the same journal since 2005. He has been an Associate Editor of Circuits, Systems, and Signal Processing since 2000. His research interests include multidimensional systems and signal processing, array signal processing, and biomedical signal processing.