

## Short Paper

---

# A Fast Mode Decision Method for H.264/AVC Using the Spatial-Temporal Prediction Scheme

CHENG-CHANG LIEN AND CHUNG-PING YU

*Department of Computer Science and Information Engineering*

*Chung Hua University*

*Hsinchu, 300 Taiwan*

In the H.264/AVC coding standard, seven motion estimation modes from  $4 \times 4$  to  $16 \times 16$  are used to find the minimum motion compensation error for each macroblock. However, the high computation cost of the full search method in the reference software JM-9.3 makes the encoding process inefficient. Therefore, the methods of applying the SAD (sum of absolute difference), homogeneous region analysis, and edge detection are developed to determine the optimum motion estimation mode. Nevertheless, the additional computation cost of image processing still reduces the efficiency of the motion compensation process. In this paper, the spatial-temporal correlations between the current frame and reference frame are analyzed to develop a fast mode decision method in which no extra image processes are used. Furthermore, the concept of drift compensation is adopted to avoid the error accumulation phenomenon during the mode decision process. The experimental results show that the computation cost may be reduced above 60% and the average PSNR is only dropped about 0.04db.

**Keywords:** H.264, JM-9.3, motion estimation mode, mode decision, spatial-temporal correlation

## 1. INTRODUCTION

Recently, the new video coding standard H.264/AVC [1, 2] is proposed by the Joint Video Team (JVT) [1] to develop a new low bit-rate video compression technology. In the JVT reference software [3], seven modes (the various kinds of block sizes) are applied to perform the motion compensation process such that the R-D cost defined in Eq. (1) is optimized.

$$J_{Mode} = D + \lambda_{Mode} \times R \quad (1)$$

where  $D$  denotes the motion compensated error of a macroblock,  $\lambda_{Mode}$  is the Lagrange multiplier, and  $R$  represents the bit-rate. In order to find the optimum motion estimation mode, we must calculate the R-D cost for each motion estimation mode in which some time-consuming processes, *e.g.*, the motion estimation, DCT transformation, and quantization, are involved. In addition, seven motion estimation modes from  $4 \times 4$  to  $16 \times 16$  are used to determine the optimum motion estimation modes within a macroblock. Hence,

---

Received August 31, 2005; revised November 24, 2005; accepted December 22, 2005.

Communicated by Jeng-Neng Hwang.

the high computation cost for the full search method used in the reference software JM-9.3 [3] make the encoding process inefficient.

Recently, many researches [4-7] addressed on the fast mode decision methods are proposed. In [4], the SAD (sum of absolute difference) between the current frame and previous frame for each macroblock is applied to evaluate what motion estimation modes are appropriate for the motion estimation process. In [5], the edge detection is applied to classify the homogeneous and non-homogeneous regions. If the macroblocks belong to the homogeneous regions then the motion estimation modes 4, 5, 6 and 7 ( $8 \times 8$ ,  $8 \times 4$ ,  $4 \times 8$  and  $4 \times 4$ ) shown in Fig. 1 are removed. Otherwise, the macroblock is divided into many sub-macroblocks of size  $8 \times 8$  and each sub-macroblock is analyzed whether it belongs to the homogenous region or not. If each  $8 \times 8$  sub-macroblock is homogenous then the motion estimation modes 5, 6 and 7 ( $8 \times 4$ ,  $4 \times 8$  and  $4 \times 4$ ) are removed. This process is continued until all sub-macroblock are evaluated. Finally the rate-distortion optimization process is used to determine which motion estimation mode is best. In [7], Zhu *et al.* applied the low-resolution image and edge detection to determine the motion estimation modes ( $8 \times 4$ ,  $4 \times 8$  and  $4 \times 4$ ). However, the efficiency of the motion compensation process will be reduced by the extra image processes for determining the appropriate motion estimation modes.

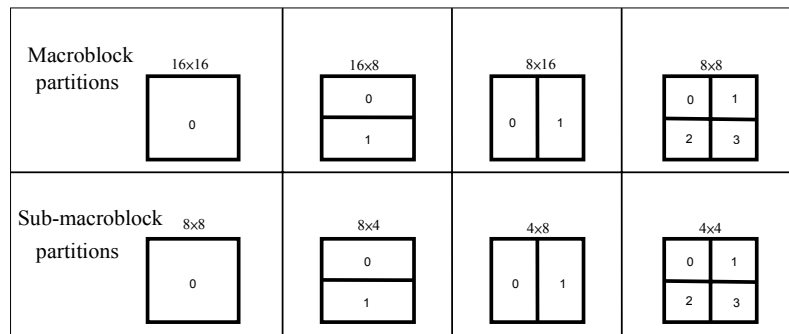


Fig. 1. Macroblock and sub-macroblock partitions in H.264.

In [6], the total energy of AC coefficients is used to evaluate the intrinsic complexity for each macroblock and then the motion estimation modes may be determined. However, summation process also reduces the computation efficiency. In this paper, the spatial-temporal correlations among the reference frames and neighboring macroblocks are analyzed to develop a fast mode decision method in which no extra image processes are used. Furthermore, the concept of drift compensation [8] is adopted to avoid the error accumulation during the mode decision process. By comparing with JM-9.3 reference software, the experimental results show that the computation cost may be reduced above 60% and the average PSNR is only dropped about 0.04db.

## 2. H.264/AVC AND JM-9.3 REFERENCE SOFTWARE

The main objectives of H.264/AVC coding standard are to improve the compression

efficiency, network friendly, error robustness and video representation for interactive and non-interactive applications. In the H.264/AVC coding standard, large flexibilities for the motion estimation modes, multiple reference frames, intra prediction mode for I-frames, motion estimation refinements, entropy coding ... *etc.* are provided to obtain the optimum R-D cost.

Generally, the H.264/AVC defines four encoding profiles: baseline, main, extended, and fidelity range extensions (FREXT) [9]. In this paper, only the baseline profile (I- and P- slices) is considered to develop the fast mode decision method. The inter-coded macroblocks in the P-slices are encoded by the motion compensated process with multiple reference frames (previous coded pictures). Then the residual data are transformed and quantized with the  $4 \times 4$  integer transformation [2]. The transform coefficients are coded by using the context-adaptive variable length coding scheme (CAVLC) [2].

In H.264/AVC, there are totally seven block sizes shown in Fig. 1 are used for the motion estimation process. Hence, the motion vectors will be estimated for each partition of macroblock and sub-macroblock. However, the full searching process ( $16 \times 16$ ,  $16 \times 8$ , ...,  $4 \times 4$ ) adopted in the JM-9.3 makes the encoding process inefficient. In this paper, the fast mode decision method is developed on the H.264/AVC reference software JM-9.3 [3].

### 3. THE FAST MODE DECISION METHOD

In this section, firstly, we will analyze the spatial-temporal mode correlations among the spatial and temporal macroblocks. Based on the spatial-temporal mode correlation, the fast mode decision method is developed.

#### 3.1 The Spatial-Temporal Mode Correlation

By the careful observation of the mode decision process in the JM-9.3, the motion estimation modes of a macroblock is highly correlated with the modes of the macro blocks neighboring to the same position on the previous reference frames (multiple reference frames). Based on the JM reference software, the mode correlation is analyzed and listed in Table 1. Here, the macroblock partitions:  $16 \times 16$ ,  $16 \times 8$ ,  $8 \times 16$ ,  $8 \times 8$ , and skip mode are used to analyze the mode correlation. Nine video sequences with length 200 frames are used to analyze the mode correlation. Furthermore, the quantization step size is set as  $QP = 28, 32, 36$  and  $40$  respectively and the skip and  $16 \times 16$  modes are regarded as the same class.

In Tables 1, we illustrate the probabilities that the estimation mode of a macroblock belongs to the top one and top two estimation modes among the neighboring macroblocks centered at the same position on the previous frame. It shows that the larger  $QP$  is, the stronger the correlation is. Moreover, we found that the motion estimation mode of a macroblock is highly correlated with the motion estimation modes of the macroblocks neighboring to the same position on the previous reference frame. According to the mode correlation analysis, we may construct the mode histogram to develop a new fast mode decision method.

**Table 1. Probabilities for the mode of a macroblock on P-frame #2 (see Fig. 2) belongs to the (a) top one and (b) top two modes sorted from the neighboring macroblocks centered at the same position on P-frame #1.**

(a)

Sequences	$QP28$	$QP32$	$QP36$	$QP40$
Akiyo	90.10%	93.78%	96.76%	98.57%
Container	89.18%	93.38%	96.46%	98.49%
Hall_Monitor	91.58%	92.65%	93.74%	95.60%
Moth&Daug	78.31%	87.71%	93.86%	97.30%
News	81.11%	85.39%	90.04%	93.76%
Salesman	85.52%	87.99%	92.64%	96.69%
Carphone	64.70%	76.14%	85.99%	92.74%
Coastgrd	53.23%	65.31%	78.79%	89.42%
Foreman	54.29%	63.22%	74.78%	84.00%

(b)

Sequences	$QP28$	$QP32$	$QP36$	$QP40$
Akiyo	93.48%	95.74%	97.76%	98.94%
Container	93.08%	95.37%	97.30%	98.70%
Hall_Monitor	96.01%	96.48%	96.41%	97.22%
Moth&Daug	86.42%	92.06%	95.95%	97.85%
News	87.86%	90.41%	93.16%	95.51%
Salesman	91.85%	92.35%	95.19%	97.80%
Carphone	78.22%	84.90%	90.51%	94.81%
Coastgrd	73.61%	79.11%	86.36%	93.32%
Foreman	72.11%	77.30%	84.31%	89.66%

### 3.2 The Fast Mode Decision Method using Spatial-Temporal Mode Correlation

For each GOP, the mode decision for the first P-frame is determined by using the full search method and the determined modes for each macroblock are used to predict the modes for the following P-frames. The algorithm of the fast mode decision method is described as follows.

**Step 1: Tracking of macroblock.** To find the accurate mode histogram for each macroblock, each macroblock on current frame should be tracked. By tracking each macroblock on previous frame with the weighted motion vectors, we may find the corresponding macroblock on the current frame. Fig. 2 illustrates the tracking process.

$$MV(x, y) = \left( \sum_{i=x-1}^{i=x+1} \sum_{j=y-1}^{j=y+1} w_{ij} \mathbf{m}(i, j) \right) \quad (2)$$

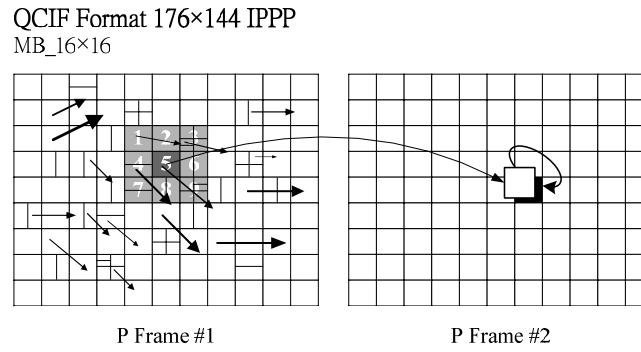


Fig. 2. Motion vectors of the marked macroblocks on P-frame #1 used to track the macroblock on P-frame #2.

where  $MV$  is the predicted motion vector,  $x$  and  $y$  denote the block coordinates of previous frame,  $m(i, j)$  is calculated from the macroblock partitions with nonzero motion vectors in block  $(i, j)$  on previous frame (number 1-9 in Fig. 2), and  $w_{ij}$  denotes the weight.

**Step 2: Calculation of the mode histogram.** Once each macroblock on current frame is tracked, the position of the tracked macroblock on previous frame shown in Fig. 2 may be found. Although, the position that the motion vector refers to may not on the MB boundary, we will choose the MB that is the nearest one (see Fig. 2). Then, the mode histogram for each tracked macroblock may be obtained by calculating the number of each mode among the neighboring macroblocks centered at the position of the tracked macroblock on previous frame. However, to reduce the computation cost, eight modes including skip mode are classified into five categories (listed in Table 2) according to their block size. Instead of calculating the mode histogram, the category histogram is calculated and sorted. Based on the sorted category histogram we select the block modes in the top two categories as the candidate modes for the mode decision process.

Table 2. Five categories for the block mode classification.

Mode Category	
1	SKIP / $16 \times 16$
2	$16 \times 8$ / $8 \times 16$
3	$8 \times 8$
4	$8 \times 4$ / $4 \times 8$
5	$4 \times 4$

**Step 3: Drift compensation.** In order to prevent the drift phenomenon in the mode decision and motion estimation processes, the candidate categories need to be refined when the R-D cost for a macroblock  $B_{ij}$  is larger than a predefined threshold  $T_{ij}$ . Firstly, we record the R-D cost  $C_{ij}$  of each macroblock  $B_{ij}$  in first P-frame obtained from the JM mode decision process (full searching). Then, the threshold value for each macroblock  $T_{ij}$  is set as  $\alpha C_{ij}$  and the R-D cost of each macroblock for the successive P-frames will be

compared to  $T_{ij}$ . If the R-D cost of macroblock  $B_{ij}$  is greater than  $T_{ij}$ , the mode decision process will be refined as the following rules:

1. Replace the mode category composed of the larger blocks with the next mode category composed of smaller blocks as the new candidate mode category. But, the new chosen mode category should not be the same as the one that is already in the candidate mode categories.
2. If the candidate mode categories can't be updated with the one with smaller blocks, then the candidate mode categories stop updating.

Here, the parameter  $\alpha$  is set equal to 1.5 by the careful observation on the drift compensation process.

**Step 4: Mode recording.** Once the motion estimation modes within each macroblock are determined, the partition scheme and corresponding motion vectors are recorded.

The block diagram of fast mode decision method using the spatial-temporal correlation is illustrated in Fig. 3.

#### 4. EXPERIMENTAL RESULTS

Here, the efficiency and rate-distortion for the proposed method are analyzed on the basis of JM-9.3 reference software and the simulation configurations are listed in Table 3.

**Table 3. Simulation configures for JM-9.3 reference software.**

Configures	Parameters
Length of video frames for the simulation	200
Number of reference frames	5
Search range for the motion estimation	16
Hadamard transform for encoding DC components	Open
Rate-distortion optimization	Open
Length of GOP	13
Number of test video sequences (QCIF format)	9

##### 4.1 The Efficiency and PSNR Analyses for Fixed $QP$ Parameters

Here the  $QP$  parameters in H.264/AVC are fixed as 28, 32, 36, and 40 respectively for the efficiency and rate-distortion analyses. The computation efficiency is analyzed with Eq. (3) and the transmission rate is analyzed with Eq. (4). The efficiency analysis is performed by computing the saving time for the proposed method to the JM full searching method and illustrated in Table 4. It is obviously that our proposed method may reduce the computation time about 63.01% for the nine video sequences. The PSNR analysis is illustrated in Table 5. The average PSNR of the fast mode decision method is decreased about 0.04dB. The bit-rate of proposed method is only increased about 2.98%.

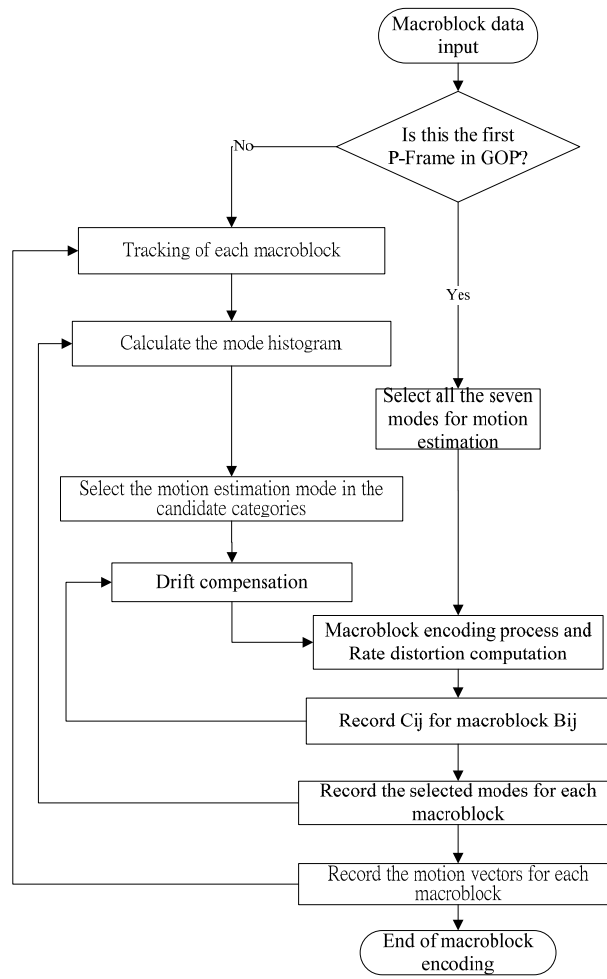


Fig. 3. Block diagram of fast mode decision method.

Table 4. Speed up analysis for three kinds of video sequences.

	Sequence	QP28	QP32	QP36	QP40
A	Akiyo	67.3%	69.2%	66.1%	66.0%
	Container	57.1%	58.3%	61.2%	60.3%
	Hall_Monitor	66.0%	66.4%	66.5%	65.9%
	Moth&Daug	61.7%	62.1%	61.3%	60.1%
B	News	63.8%	63.4%	64.7%	63.3%
	Salesman	66.3%	66.5%	64.4%	62.0%
C	Carphone	62.7%	62.7%	62.3%	63.2%
	Coastgrd	64.1%	63.6%	62.3%	60.4%
	Foreman	57.5%	59.0%	60.5%	60.6%

A: Low motion sequences B: Regular motion sequences C: High motion sequences

**Table 5. PSNR (dB) analysis.**

	Sequence	QP28	QP32	QP36	QP40
A	Akiyo	-0.04	-0.04	-0.03	-0.04
	Container	-0.03	-0.03	-0.03	-0.02
	Hall_Monitor	0	-0.01	0.02	-0.07
	Moth&Daug	-0.06	-0.06	-0.07	-0.01
B	News	-0.04	-0.06	-0.05	-0.04
	Salesman	-0.01	-0.01	-0.03	-0.01
C	Carphone	-0.07	-0.1	-0.13	-0.06
	Coastgrd	-0.05	-0.05	-0.07	-0.05
	Foreman	-0.05	-0.1	-0.12	-0.08

In general, the degree of the efficiency improvement for the video sequences with high motion (Carphone, Coastgrd, Foreman) is less than the videos with low motion (Akiyo, Container, Hall\_Monitor, Moth&Daug).

$$\Delta T = \frac{Time[JM] - Time[proposed]}{Time[JM]} \times 100\% \quad (3)$$

$$\Delta R = \frac{BitRate[proposed] - BitRate[JM]}{BitRate[proposed]} \times 100\% \quad (4)$$

Furthermore, the number of excluded modes during the mode decision process is computed. For the video sequence "Foreman" with length 200 and 99 macroblocks for each frame, the total number of mode searching in JM-9.3 is about 115284 and the number in our proposed method is about 53782. The percentage of excluded modes is about 53.4%.

#### 4.2 The Rate-Distortion Analyses

The rate-distortion is analyzed by applying the R-D control scheme in the JM-9.3 reference software. For each bit-rate, the parameter  $\lambda$  in Eq. (1) had been determined in the JM-9.3 to optimize the R-D cost. The rate-distortion analyses are performed with the following two configure settings: (1) All the motion estimation modes in the JM-9.3 are used, (2) only the motion estimation modes:  $16 \times 16$ ,  $16 \times 8$ , and  $8 \times 16$  are used. From Fig. 4, the simulation results show that the PSNR value is closed to the optimal value obtained from the JM-9.3 reference software. From Fig. 5, our proposed fast mode decision method may improve the computation cost greatly. Finally, we compare the encoding time saving and PSNR to the other fast mode decision methods. All the methods are compared to the JM9.3 reference software using the efficiency and PSNR analyses. Tables 6 and 7 show the comparisons for the proposed method to the methods with extra image processing (Wu's method [5] and Jing's method [4]). In addition, Table 8 illustrates the comparisons for the proposed method to the method without extra image processing (Yu's method [6]). It is obvious that our method may improve the efficiency about 20%-30%. Furthermore, the PSNR performance outperforms the other methods.

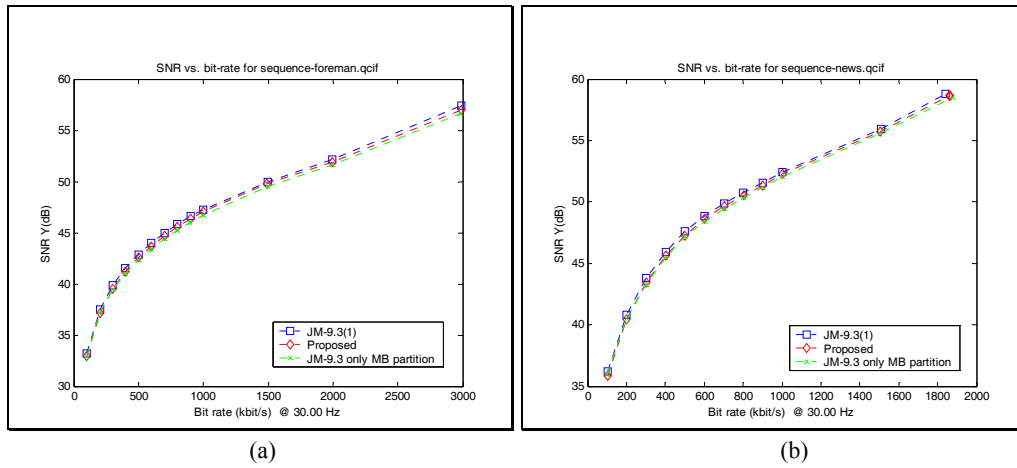


Fig. 4. Rate-distortion curves for (a) Foreman and (b) News video sequence obtained from JM-9.3, our proposed method, and JM-9.3 with only Macroblock partition.

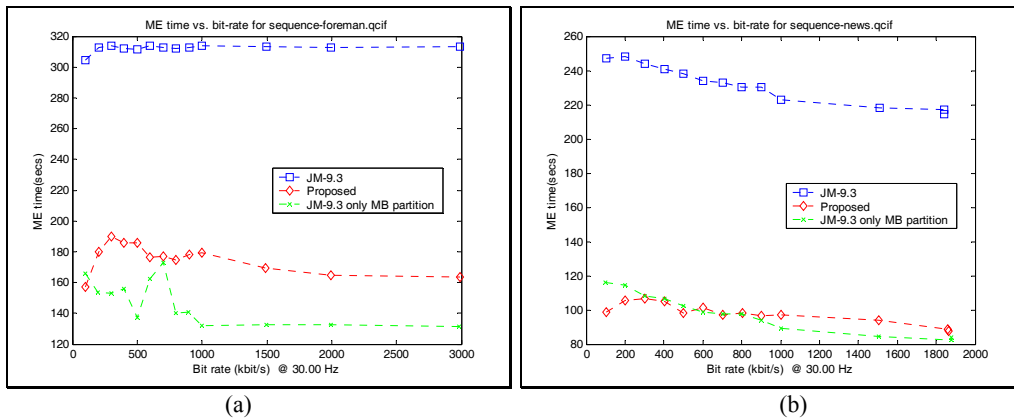


Fig. 5. Computation times of the motion estimation process for (a) Foreman and (b) News sequences using the JM-9.3 reference software, our proposed method, and JM-9.3 with only Macroblock partition.

Table 6. Efficiency and PSNR comparisons for the proposed method and Wu’s method [5].

Sequence	Encoding time saving (%)		PSNR decrease (dB)	
	Wu’s method	Proposed	Wu’s Method	Proposed
Foreman	18%	48%	- 0.09	- 0.04
News	39%	65%	- 0.15	- 0.03
Container	38%	60%	- 0.08	- 0.04

Table 7. Efficiency and PSNR comparisons for the proposed method and Jing’s method [4].

Sequence	Encoding time saving (%)		PSNR decrease (dB)	
	Jing’s method	Proposed	Jing’s method	Proposed
Carphone	36%	56%	- 0.08	- 0.08
Moth&Daug	39%	64%	- 0.07	- 0.05

**Table 8. Efficiency and PSNR comparisons for the proposed method and Yu's method [6].**

Sequence	Encoding time saving (%)		PSNR decrease (dB)	
	Yu's method	Proposed	Yu's method	Proposed
Akiyo	29%	67%	- 0.03	- 0.04
Forman	25%	57%	- 0.09	- 0.05
Coastgrd	29%	64%	- 0.11	- 0.05

## 5. CONCLUSION

In this paper, the spatial-temporal correlations between the current frame and the reference frame are considered to develop a fast mode decision method in which no extra image processes are used. Furthermore, the concept of drift compensation is adopted to avoid the error accumulation phenomenon during the mode decision process. The experimental results show that the computation cost may be reduced about 60% and average PSNR is only dropped about 0.04db.

## REFERENCES

1. Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264/ISO/IEC 14486-10 AVC) in Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, JVT-G050, 2003.
2. I. E. G. Richardson, *H.264 and MPEG-4 Video Compression*, John Wiley & Sons, 2003.
3. Joint Video Team (JVT) Reference software JM-9.3, <http://iphome.hhi.de/suehring/tml/>, 2005.
4. X. Jing and L. P. Chau, "An efficient inter mode decision approach for H.264 video coding," in *Proceedings of IEEE International Conference on Multimedia and Exposition*, Vol. 2, 2004, pp. 1111-1114.
5. D. Wu, S. Wu, K. P. Lim, P. Feng, Z. G. Li, and C. C. Ko, "Fast inter mode decision with adaptive thresholds for H.264 encoding," in *Proceedings of the 8th International Symposium on Consumer Electronics*, 2004, pp. 406-409.
6. A. C. Yu, "Efficient block-size selection algorithm for inter-frame in H.264/Mpeg-4 AVC," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, Vol. 3, 2004, pp. 169-172.
7. D. Zhu, Q. Dai, and R. Ding, "Fast inter prediction mode decision for H.264," in *Proceedings of the 5th IEEE International Conference on Multimedia and Exposition*, Vol. 2, 2004, pp. 1123-1126.
8. P. Yin, A. Vetro, H. Sun, and B. Liu, "Drift compensation for reduced spatial resolution transcoding," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 12, 2002, pp. 1009-1020.
9. G. J. Sullivan, P. Topiwala, and A. Luthra, "The H.264/AVC advanced video coding standard: Overview and introduction to the fidelity range extensions," in *Proceedings of SPIE Conference on Applications of Digital Image Processing*, Vol. 5558, 2004, pp. 454-474.

**Cheng-Chang Lien (連振昌)** was born in Taoyuan, Taiwan, R.O.C., in 1964. He received the B.S. degree from the Electrical Engineering Department, Chung Yuan University, Taoyuan, Taiwan, in 1987. He received the M.S. and Ph.D. degrees from the Electrical Engineering Department, National Tsing Hua University, Hsinchu, Taiwan, in June 1992 and September 1997, respectively. Currently, he is an assistant professor in Computer Science and Information Engineering Department, Chung Hua University, Hsinchu, Taiwan. His research interests are in the area of computer vision and image/visual signal processing.

**Chung-Ping Yu (喻仲平)** was born in Taipei, Taiwan, R.O.C., in 1981. He received the M.S. degree from the Department of Computer Science & Information Engineering, Chung Hua University, Hsinchu, Taiwan, in July 2005. His research interests are in the area of video compression and image processing.