Fast Mode Decision Based on Mode Adaptation

Tiesong Zhao, *Student Member, IEEE,* Hanli Wang, *Member, IEEE,* Sam Kwong, *Senior Member, IEEE,* and C.-C. Jay Kuo, *Fellow, IEEE*

***Abstract*—This paper proposes an efficient algorithm for fast mode decision in H.264/advanced video coding by adaptively predicting the optimal mode for each macroblock (MB) to be coded. Firstly, encoding modes are projected as points onto a 2-D map, and an optimal 2-D point of the MB to be coded is predicted based on the encoding information of spatial–temporal neighboring blocks. Then, a priority-based mode candidate list with a descending order to be the best mode is constructed based on the optimal 2-D point. Finally, mode decision is performed according to the priority-based mode candidate list in the checking order, from the most important mode to the least one, with early termination conditions. Extensive experimental results demonstrate that the proposed algorithm is superior to three recent fast mode decision algorithms, with the entire encoding time being reduced by about 60% for quarter common intermediate format/common intermediate format/standard-definition sequences on average and the rate distortion performance being kept almost intact.**

***Index Terms*—2-D map, H.264/advanced video coding (AVC), mode decision, spatial–temporal neighboring prediction.**

I. Introduction

With more advanced coding techniques adopted, the state-of-the-art video coding standard H.264/advanced video coding (AVC) [1] achieves significant coding efficiency as compared with previous video coding standards. However, the computational complexity for H.264/AVC encoding is dramatically increased. It is important to reduce H.264/AVC encoding computations for real-world applications. In H.264/AVC, the rate distortion optimization (RDO)-based mode decision employs more image partitions as well as more sophisticated sub-pixel motion search techniques and, consequently, makes compression more efficient. However, these new features consume more computational power. In general, the mode decision process with motion estimation (ME) would take about 90% of the overall encoding time [2].

In recent years, many researchers have focused on fast algorithms to simplify the mode decision process. In [3], a fast mode decision algorithm called monotonic error surface-based prediction (MESBP) was proposed, which makes use of the rate distortion (RD) costs of INTER16 × 16, INTER8 × 8, and INTER4 × 4 to determine whether the other INTER modes should be checked or skipped, with a fixed checking order. Lee *et al.* [4] developed a fast mode decision scheme by classifying coding modes based on RD costs. The RD costs of coding modes, which are already checked, are used to decide whether INTRA modes and some INTER modes could be skipped. In [5], the mean absolute difference of a macroblock (MB) was employed to skip redundant sub-MB INTER modes (i.e., the MB partition size is equal to or less than 8 × 8). Choi *et al.* [6] designed an algorithm for early SKIP mode decision and selective INTRA mode decision. In this algorithm, for each MB of P and B slices, the SKIP mode is checked first. If some predefined early termination condition holds, the SKIP mode is considered as the best and all other modes are consequently skipped. In [7], a fast INTER mode decision algorithm was proposed to skip redundant modes by making use of spatial and temporal characteristics, where both spatial homogeneity (which is based on MB's edge intensity) and temporal stationarity (which is based on differences between the current MB and its co-located block in the reference frame) were employed. Experimental results showed that the algorithm of [7] could reduce about 30% of the entire encoding time on average when it was implemented into JM5.0 [8]. Kim *et al.* [9] proposed to use all-zero block early detection techniques for elimination of unnecessary modes. In [10], Wang *et al.* proposed an algorithm for fast mode decision, which comprises early termination of mode decision and ME (which is based on the sufficient condition for all-zero block detection [11]), temporal–spatial checking, threshold-based prediction, and MESBP. However, the temporal–spatial correlations of MB coding modes are not fully investigated in [10]. In [12], a hierarchical algorithm was developed for fast mode decision. In particular, the INTER modes of an MB were categorized into three levels: the SKIP mode at the first level, INTER16 × 16, INTER16 × 8, and INTER8 × 16 modes at the second level, and the remaining sub-MB INTER modes at the third level. At each level, different strategies were implemented for the purpose of fast mode decision. Liu *et al.* [13] proposed to use the motion vector (MVs) of 4 × 4 blocks for fast mode decision. In [13], the INTER4 × 4

Manuscript received March 9, 2009; revised August 14, 2009. First version published March 15, 2010; current version published May 5, 2010. This work was supported by the Hong Kong Research Grants Council General Research Fund, under Projects 9 041 236 (CityU 114 707) and 9 041 353 (CityU 115 408), and the German Alexander von Humboldt Research Fellowship. Joint first authors contributed equally to this work. This paper was recommended by Associate Editor H. Yu.

T. Zhao and S. Kwong are with Department of Computer Science, City University of Hong Kong (CityU), Kowloon, Hong Kong (e-mail: ztiesong2@student.cityu.edu.hk; cssamk@cityu.edu.hk).

H. Wang is with the Department of Computer Science, Tongji University, Shanghai 201804, China (e-mail: hanli.wang@ieee.org).

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

Digital Object Identifier 10.1109/TCSVT.2010.2045812

mode was checked first, and the resultant 4 × 4 blocks' MVs were utilized to skip checking redundant modes given a predesigned rule. However, for natural video sequences, there is a very low probability that an MB is encoded with INTER4 × 4 as the best mode [2]. Therefore, it is not very economical to check the INTER4 × 4 mode first. In [14], Ri *et al*. combined spatial–temporal mode prediction and RD cost prediction together, in which the spatially and temporally predicted modes derived from neighboring MBs would be firstly checked; if the minimum RD cost of the two is less than the predicted RD cost multiplied by a constant value, then the other inter-modes could be skipped. Another algorithm proposed by Zeng *et al.* [15] comprised both RD cost-based early skip mode checking and motion activity-based mode classification, and the encoding time could be significantly reduced.

Although fast mode decision algorithms are well designed, there is still a need to further develop more efficient algorithms, especially for high-resolution video contents such as the standard-definition (SD) video and high-definition (HD) video. In this paper, an efficient mode decision algorithm is designed based on novel mode adaptation techniques. For an MB to be encoded, the coding modes of spatial–temporal neighboring blocks are firstly projected as mode points onto a 2-D map. Based on the projected mode points in the 2-D map, an optimal central point is deduced, and a priority-based mode candidate list is built according to the distances between all INTER mode points and the deduced central point. Then, mode decision is carried out based on the mode candidate list in the following order: from the most important mode (which is corresponding to the nearest mode point from the central point) to the least important mode (which is corresponding to the farthest mode point from the central point). During the mode decision process, two conditions are used for early termination purpose.

The rest of this paper is organized as follows. In Section II, the mode adaptation method is introduced and discussed in detail. The overall fast mode decision algorithm is presented in Section III. Experimental results are provided in Section IV. Finally, Section V concludes this paper.

II. Optimal Mode Adaptation

A. Review of Previous Work

In video coding, it is common that temporal or spatial neighboring MBs tend to have the same or similar coding mode. This is because there are many homogenous or stationary regions in natural video sequences. Therefore, it is desired to check those similar modes first and, if some early termination condition is satisfied, the mode decision process can be stopped. Based on this observation, the authors developed a 2-D map-based method [16] to construct an optimal mode candidate list for fast mode decision. In this method, the coding modes of spatial–temporal neighboring MBs are projected as mode points onto a 2-D map with their coordinates in proportion to its MB or sub-MB partition size. Then, an optimal point in the 2-D map is deduced by simply averaging the coordinates of the projected mode points. According to the distances between the optimal point and all INTER mode points, a mode candidate list is constructed with associated coding modes in an ascending order of the distances. After the mode candidate list is available, the mode decision process is run by checking modes of the list in order. When combined with early termination techniques, the computations consumed by the mode decision process are greatly reduced.

In this paper, the 2-D map-based method [16] is further improved in the following two aspects. Firstly, in Section II-B, instead of mapping a coding mode only according to its partition size, it is fully investigated to efficiently convert a coding mode to a mode point in the 2-D map. Secondly, in Subsection II-C, spatial–temporal adaptive weights are introduced to deduce the optimal point for building the priority-based mode candidate list.

B. Mapping Coding Modes to 2-D Mode Points

In order to construct the priority-based mode candidate list for the current MB to be coded, the coding modes of spatial–temporal neighboring MBs are utilized by being mapped to mode points in a plane. For computing the location of coding mode points, the normalized motion vector (NMV) of a 4×4 block is firstly defined as

$$\mathrm{NMV}_{x,y} = \begin{cases} \frac{\mathrm{MV}_{x,y}}{|c-r_0|}, & \text{if } pdir = 0 \\ \frac{\mathrm{MV}_{x,y}}{|c-r_1|}, & \text{if } pdir = 1 \\ \frac{1}{2}\left(\frac{\mathrm{MV}_{x,y}}{|c-r_0|} + \frac{\mathrm{MV}_{x,y}}{|c-r_1|}\right), & \text{if } pdir = 2 \end{cases} \tag{1}$$

where x and y are the horizontal and vertical locations of a 4×4 block, respectively; $\mathrm{MV}_{x,y}$ is the MV of the 4×4 block with block index (x, y); $pdir$ indicates the prediction direction, which may take values of 0, 1, and 2 for forward, backward, and bi-direction prediction, respectively; c, r_0, and r_1 indicate the current frame index, the reference frame index in list 0 and the reference frame index in list 1, respectively.

Then, the mode point location characterized by (H_h, H_v), including the horizontal component H_h and vertical component H_v, of an MB or sub-MB with the upper-left 4 × 4 block location at $(x0, y0)$ is defined as

$$\begin{aligned} H_h &= \frac{1}{N}\sum_{k=0}^{N-1} f_\sigma\left(\{\mathrm{NMV}_{x0+i,y0+k} \mid i = 0\cdots N-1\}\right) \\ H_v &= \frac{1}{N}\sum_{k=0}^{N-1} f_\sigma\left(\{\mathrm{NMV}_{x0+k,y0+i} \mid i = 0\cdots N-1\}\right) \end{aligned} \tag{2}$$

where N denotes the number of 4 × 4 blocks in a row or a column of an MB ($N = 4$) or a sub-MB ($N = 2$), and function $f_\sigma(\cdot)$ is used to calculate the standard variance; namely

$$\begin{aligned} &f_\sigma(\{\mathrm{NMV}_{x0+i,y0+k} \mid i = 0\cdots N-1\}) \\ &= \frac{1}{N}\sum_{i=0}^{N-1}\left\|\mathrm{NMV}_{x0+i,y0+k} - \frac{1}{N}\sum_{j=0}^{N-1}\mathrm{NMV}_{x0+j,y0+k}\right\|^2 \end{aligned} \tag{3}$$

where $\|\cdot\|$ is the inner product operator to calculate the Euclidean distance between two MV points.

TABLE I
MODE POINT LOCATION (H_h, H_v) OF INTER MODES FOR *Foreman* (QCIF)

Mode	$Q_p = 24$	$Q_p = 28$	$Q_p = 32$
DIRECT/SKIP	(0.00, 0.00)	(0.00, 0.00)	(0.00, 0.01)
16 × 16	(0.00, 0.00)	(0.00, 0.00)	(0.00, 0.00)
16 × 8	(0.00, 1.45)	(0.00, 1.93)	(0.00, 2.05)
8 × 16	(1.55, 0.00)	(1.81, 0.00)	(2.20, 0.00)
P8 × 8	(1.90, 1.84)	(2.24, 2.16)	(2.49, 2.42)
8 × 8	(0.00, 0.00)	(0.00, 0.00)	(0.00, 0.00)
8 × 4	(0.00, 1.40)	(0.00, 1.65)	(0.00, 1.94)
4 × 8	(1.32, 0.00)	(1.53, 0.00)	(1.70, 0.00)
4 × 4	(1.55, 1.37)	(2.13, 1.94)	(3.02, 2.43)

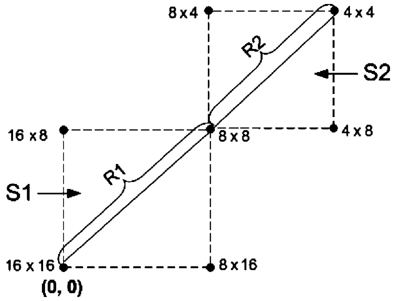


Fig. 1. Illustration of mode points with the origin (0, 0) located at the mode point of INTER16 × 16.

By collecting the location values of mode points on various video sequences, it is observed that the mode points of INTER16 × 16, INTER16 × 8, INTER8 × 16, and P8 × 8 (i.e., all sub-MB INTER modes) form an approximate square in a plane, and the similar square-shape observation is also applicable for the four sub-MB INTER modes, including INTER8 × 8, INTER8 × 4, INTER4 × 8, and INTER4 × 4. It should be mentioned that, for calculating the location of mode points with (2), the modes DIRECT/SKIP, INTER16 × 16, INTER16 × 8, INTER8 × 16, and P8 × 8 are computed at the MB level with $N = 4$, and INTER8 × 8, INTER8 × 4, INTER4 × 8, and INTER4 × 4 are computed at the sub-MB level with $N = 2$. The statistics of larger Q_p values are discarded due to less amount of sub-MB modes as the best mode. In fact, there are very few MBs to be coded with INTER4 × 4 even for $Q_p = 32$ and, therefore, the deviation of the mode point location for INTER4 × 4 is too small to apply. Taking the video sequence *Foreman* [quarter common intermediate format (QCIF)] as an example, the location values of mode points for different INTER modes are listed in Table I.

In general, the illustration of mode points can be shown in Fig. 1, where two approximate squares $S1$ and $S2$ are formed as mentioned above. Note that the origin (0, 0) of the mode point plane is set as located at the mode point of INTER16 × 16; and the two squares $S1$ and $S2$ can be characterized by the corresponding diagonal $R1$ and $R2$, respectively. In the proposed work, it is necessary to study the relation between the positions of $S1$ and $S2$, which can be converted to study the relative scale $R1/R2$ of these two squares. In order to find the best scale value $R1/R2$, we design eight typical parameter sets as given in Table II, where a typical mode point location is specified for each INTER mode. These eight mode mapping parameter sets are individually implemented in our previous work [16] for comparison. The mode mapping parameter set that produces the best video performance will be chosen and used in this paper. The average encoding results are also presented in Table II in terms of the peak-signal-to-noise-ratio decrease (ΔPSNR, dB) and the bitrate increase (ΔBR, %) as compared with the original encoder JM13.2 [8] as

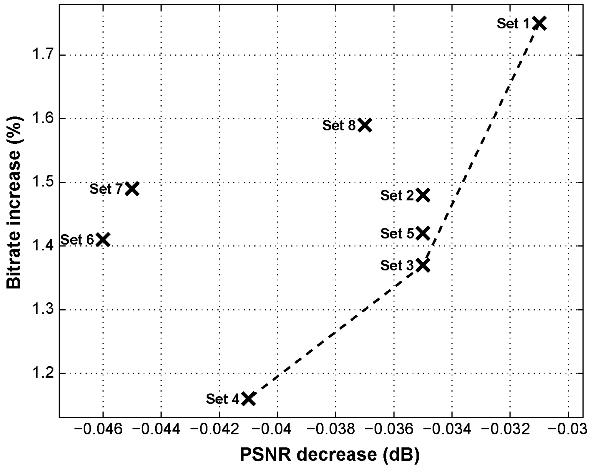


Fig. 2. RD performance of the eight mode mapping parameter sets.

$$\Delta\text{PSNR} = \text{PSNR}_p - \text{PSNR}_o \tag{4}$$

$$\Delta\text{BR} = \frac{\text{BR}_p - \text{BR}_o}{\text{BR}_o} \times 100\% \tag{5}$$

where PSNR_p and BR_p are the performance results of the algorithm [16]; PSNR_o and BR_o are the performance results of the original encoder JM13.2 [8]. The performance in terms of entire encoding time saving is not given since it keeps almost the same with different mode mapping parameter sets. Five QCIF sequences (*Coastguard*, *Foreman*, *Mobile*, *News*, and *Silent*) and three common intermediate format (CIF) sequences (*Football*, *Funfair*, and *Table Tennis*) with different Q_p values (from 24 to 40) and two kinds of group of picture (GOP) structures (IPPP and IBBP) are evaluated.

The RD performance of the eight mode mapping parameter sets is drawn in Fig. 2. From Table II and Fig. 2, it can be observed that Sets 1, 3, and 4 are in the Pareto-optimal front [17]. That is, Sets 1, 3, and 4 are nondominant with each other, and none of the other five sets are better than Sets 1, 3, and 4 in terms of both PSNR and BR. In this paper, we choose Set 4 as the best mode mapping parameter set, since the bitrate performance of Set 4 is significantly better than Sets 1 and 3 while the PSNR performance of these three sets is not very different. Therefore, the final mode mapping parameter set is given in Table III, where the SKIP/DIRECT mode is mapped to the same point as the INTER mode with the same partition size since they tend to have a similar mode point location as indicated in Table I. As for INTRA modes, they could be mapped to the INTER modes with the same partition size, too, which is shown to be an acceptable solution in experimental

TABLE II
COMPARISON OF EIGHT MODE MAPPING PARAMETER SETS

Mode Mapping Parameters								Results	
Set	16 × 16	16 × 8	8 × 16	8 × 8	8 × 4	4 × 8	4 × 4	ΔPSNR	ΔBR
1	(0, 0)	(0, 1)	(1, 0)	(1, 1)	(1, 3)	(3, 1)	(3, 3)	−0.031	1.75
2	(0, 0)	(0, 2)	(2, 0)	(2, 2)	(2, 5)	(5, 2)	(5, 5)	−0.035	1.48
3	(0, 0)	(0, 1)	(1, 0)	(1, 1)	(1, 2)	(2, 1)	(2, 2)	−0.035	1.37
4	(0, 0)	(0, 3)	(3, 0)	(3, 3)	(3, 5)	(5, 3)	(5, 5)	−0.041	1.16
5	(0, 0)	(0, 2)	(2, 0)	(2, 2)	(2, 3)	(3, 2)	(3, 3)	−0.035	1.42
6	(0, 0)	(0, 2)	(2, 0)	(3, 3)	(3, 5)	(5, 3)	(5, 5)	−0.046	1.41
7	(0, 0)	(0, 2)	(2, 0)	(3, 3)	(4, 6)	(6, 4)	(6, 6)	−0.045	1.49
8	(0, 0)	(0, 2)	(2, 0)	(2, 2)	(3, 5)	(5, 3)	(5, 5)	−0.037	1.59

TABLE III
MODE MAPPING PARAMETER SET

Mode	Location (2-D value)
SKIP, DIRECT16 × 16, INTER16 × 16, INTRA16 × 16	(0, 0)
INTER16 × 8	(0, 3)
INTER8 × 16	(3, 0)
DIRECT8 × 8, INTER8 × 8, INTRA8 × 8	(3, 3)
INTER8 × 4	(3, 5)
INTER4 × 8	(5, 3)
INTER4 × 4, INTRA4 × 4	(5, 5)

results. Given the mode mapping parameter set in Table III, an MB can be projected to a plane (that is referred to as the *2-D map* for the rest of this paper) by averaging the location values of its four 8 × 8 blocks. The resultant location is denoted as the *2-D value* of that MB.

C. Construction of Mode Candidate List With Spatial–Temporal Adaption

As stated in Section II-A, to construct the priority-based mode candidate list for fast mode decision, an optimal central point in the 2-D map should be firstly determined. In natural video sequences, there are strong correlations among coding modes of spatial–temporal neighboring MBs. Therefore, the 2-D values of spatial–temporal neighboring MBs could be used to deduce the 2-D value of the central point for the current MB being coded.

For the current MB, there are at most 22 spatial–temporal neighboring 8 × 8 blocks (6 spatial blocks in the current frame and 16 temporal blocks in the previous frame). Let $M(n, x, y)$ be the upper-left 8 × 8 block of the current MB that is in the nth frame with the upper-left pixel at (x, y) and $M(m, i, j)$ be the spatial–temporal neighboring 8 × 8 blocks as summarized in Table IV, where $\Delta n = m - n$, $\Delta x = i - x$, and $\Delta y = j - y$.

Based on observations on various video sequences, we find that it is efficient to use a partial set of the spatial–temporal 8 × 8 blocks to derive the optimal central point for the current MB. For description convenience, we take the video sequence *News* (QCIF with IBBP structure) for an example. Similar observations are applicable to other video sequences. In Table IV, the distances of 2-D values between the current MB and its neighboring 8 × 8 blocks are statistically given under different Q_p conditions.

Several conclusions can be drawn from Table IV. Firstly, all distances are very small as compared to the maximal distance of 2-D map (i.e., $5\sqrt{2}$, which is the distance between the 2-D value of INTER16 × 16 and the 2-D value of INTER4 × 4 in Table III). Therefore, it is reasonable to employ spatial–temporal neighboring blocks to predict the best mode for the current MB. Secondly, all distances become shorter with increase in Q_p, which indicates that the correlations of coding modes among spatial–temporal neighboring blocks become stronger for lower bitrate coding. Thirdly, the distances of the first 10 spatial–temporal neighboring blocks (with index from 0 to 9 in Table IV) are smaller than the others. Inspired by this, only these 10 spatial–temporal neighboring blocks are utilized for deriving the optimal central point in this paper.

For these 10 neighboring blocks, two sets are defined. One is the spatial set (denoted as S_s) that includes the spatial neighboring 8 × 8 blocks with index from 0 to 5. The other is the temporal set (denoted as T_s) that includes the temporal neighboring 8 × 8 blocks with index from 6 to 9. It is obvious that blocks in T_s are better to be used for prediction in static background regions while blocks belonging to S_s are more likely to be employed in fast-moving regions. In consequence, the 2-D values of two optimal points P_S and P_T are derived as[1]

$$V_S = \frac{1}{N_S} \sum_{i=0}^{N_S-1} V_{i,S} \tag{6}$$

$$V_T = \frac{1}{N_T} \sum_{i=0}^{N_T-1} V_{i,T} \tag{7}$$

where V_S stands for the 2-D value of point P_S, N_S is the number of available 8 × 8 blocks in S_s, $V_{i,S}$ is the 2-D value of block i belonging to S_s, V_T is P_T's 2-D value, N_T is the number of available 8 × 8 blocks in T_s, and $V_{i,T}$ is the 2-D value of block i in T_s. The 2-D value of the final optimal central point P^* for the current MB is deduced as

$$V^* = w_S \cdot V_S + (1 - w_S) \cdot V_T \tag{8}$$

where w_S is the weight for point P_S and $0 \le w_S \le 1$. After P^* is determined, the priority-based mode candidate list is constructed in an ascending order of the distances between the 2-D values of all INTER modes and V^*, i.e., the nearest

[1] A 2-D point has its horizontal and vertical components. In this paper, all the mathematical operations on 2-D points are actually performed individually on both the horizontal and vertical components.

TABLE IV

MODE DISTANCE BETWEEN AN MB AND ITS SPATIAL—TEMPORAL NEIGHBORING 8 × 8 BLOCKS

Neighboring Blocks $M(m, i, j)$				Distances in 2-D Map for *News* (QCIF)				
Index	Δn	Δx	Δy	$Q_p = 24$	$Q_p = 28$	$Q_p = 32$	$Q_p = 36$	$Q_p = 40$
0	0	−8	−8	1.388	1.226	0.948	0.642	0.405
1	0	0	−8	1.215	1.094	0.833	0.571	0.253
2	0	8	−8	1.237	1.098	0.840	0.567	0.351
3	0	16	−8	1.466	1.332	1.015	0.677	0.430
4	0	−8	0	1.218	1.080	0.860	0.606	0.398
5	0	−8	8	1.245	1.123	0.873	0.602	0.391
6	−1	0	0	1.502	1.306	1.052	0.775	0.537
7	−1	8	0	1.490	1.306	1.047	0.769	0.536
8	−1	0	8	1.505	1.289	1.043	0.768	0.535
9	−1	8	8	1.506	1.301	1.046	0.778	0.538
10	−1	−8	−8	1.795	1.538	1.208	0.854	0.558
11	−1	0	−8	1.714	1.467	1.139	0.826	0.546
12	−1	8	−8	1.729	1.481	1.154	0.827	0.543
13	−1	16	−8	1.868	1.621	1.264	0.891	0.585
14	−1	−8	0	1.703	1.459	1.177	0.850	0.577
15	−1	16	0	1.754	1.532	1.213	0.882	0.592
16	−1	−8	8	1.730	1.509	1.189	0.852	0.569
17	−1	16	8	1.753	1.523	1.202	0.880	0.590
18	−1	−8	16	1.691	1.490	1.178	0.847	0.558
19	−1	0	16	1.570	1.415	1.106	0.818	0.543
20	−1	8	16	1.551	1.394	1.112	0.810	0.547
21	−1	16	16	1.696	1.533	1.187	0.868	0.572

TABLE V

PROBABILITY OF BMI IN THE MODE CANDIDATE LIST

Video	BMI in Mode Candidate List						
	0	1	2	3	4	5	6
Mother & Daughter	84.75%	5.63%	4.19%	4.52%	0.40%	0.28%	0.23%
Mobile	40.89%	16.49%	11.59%	15.48%	5.16%	5.01%	5.38%
Dancer	77.96%	7.42%	5.87%	5.20%	1.09%	1.38%	1.08%
Table Tennis	60.53%	14.58%	10.07%	9.68%	1.79%	1.82%	1.53%

mode point to the central point will be checked first; then, the second nearest mode point, and so on.

To improve the robustness of optimal central point prediction in (8), we adaptively adjust weight w_S by utilizing the coding information of the co-located MB in previous frames. Let L be the current frame index, T be a predefined number of frames that are preceding the current frame in coding order, $U(i)$ be the 2-D value of the best mode for the co-located MB in the ith frame, $V^*(i)$ be the derived 2-D value for the co-located MB in the ith frame, w_S for the current MB can be derived as

$$w_S = \arg\min_{w_S} \left\{ \frac{1}{T} \sum_{i=L-T}^{L-1} \left\| U(i) - V^*(i) \right\|^2 \right\}. \tag{9}$$

Substituting (8) into (9) and after manipulations, w_S can be rewritten as

$$w_S = \frac{\sum\limits_{i=L-T}^{L-1} [U(i) - V_T(i)] \cdot [V_S(i) - V_T(i)]}{\sum\limits_{i=L-T}^{L-1} \| V_S(i) - V_T(i) \|^2}. \tag{10}$$

Then, w_S is further refined by separating the contribution of the co-located MB in the nearest previous frame, i.e., the $(L-1)$th frame, from the other temporally co-located MBs. Firstly, two terms are defined as

$$\alpha_L = \frac{\| V_S(L-1) - V_T(L-1) \|^2}{\sum\limits_{i=L-T}^{L-1} \| V_S(i) - V_T(i) \|^2} \tag{11}$$

$$w_L = \frac{[U(L-1) - V_T(L-1)] \cdot [V_S(L-1) - V_T(L-1)]}{\| V_S(L-1) - V_T(L-1) \|^2} \tag{12}$$

where α_L and w_L are associated with the contribution of the co-located MB in the $(L-1)$th frame to calculating w_S. Considering all the other $(T-1)$ temporally co-located MBs, the derived weight can be expressed as

$$w_O = \frac{\sum\limits_{i=L-T}^{L-2} [U(i) - V_T(i)] \cdot [V_S(i) - V_T(i)]}{\sum\limits_{i=L-T}^{L-2} \| V_S(i) - V_T(i) \|^2}. \tag{13}$$

Finally, the weight for the current MB is determined as

$$w'_S = \begin{cases} w_O, & \text{if } V_S(L-1) = V_T(L-1) \\ (1-\alpha_L)\cdot w_O + \alpha_L \cdot w_L, & \text{otherwise} \end{cases} \tag{14}$$

and clipped to a valid value as

$$w_S = \min\left\{1, \max\left\{0, w'_S\right\}\right\}. \tag{15}$$

In practice, we further simplify the weight calculation by setting α_L as a constant. A typical value of α_L is 0.1. In addition, for each MB in a frame, the initial value of w_S is set as

$$w_S^0 = \beta \cdot \frac{N_S/A_S}{(N_S+N_T)/(A_S+A_T)} \tag{16}$$

where β is a regulating factor between 0 and 1, N_S and N_T are given in (6) and (7), A_S and A_T are the maximum number of the spatial and the temporal neighboring 8 × 8 blocks, respectively. In this paper, it is obvious that $A_S = 6$ and $A_T = 4$. Based on experiments, β is chosen to be 0.8. Therefore, we can rewrite (16) as

$$w_S^0 = \frac{4 \cdot N_S}{3 \cdot (N_S + N_T)}. \tag{17}$$

III. Overall Fast Mode Decision Algorithm

It is observed from exhaustive statistical analysis that it is only necessary to check a subset of modes instead of the entire priority-based mode candidate list given a distance threshold. To verify this, the probability of the best mode index (BMI) that describes the position of the best mode in the mode candidate list is given in Table V for four benchmark video sequences: *Mother & Daughter* (QCIF, IBBP), *Mobile* (QCIF, IPPP), *Dancer* (CIF, IPPP), and *Table Tennis* (CIF, IBBP) by setting Q_p values as 28/28/30 for I/P/B slices, respectively.

As shown in Table V, we see that the best mode has a very high probability within a small area in the 2-D map with the center at the predicted optimal central point. Thus, we can discard the modes which are far away from the optimal central point for fast mode decision. To this aim, a checking radius r_c is defined for *distance checking*: if the distance between a mode point and the optimal central point is larger than r_c, then we can skip checking this mode. After distance checking, those INTER modes that remain for mode decision are referred to as the set of valid INTER modes (SVIM). In this paper, we set $r_c = 3.5$.

On the other hand, since coding modes in the priority-based mode candidate list are arranged in an ascending order of distances between the corresponding mode points and the optimal central point, it is highly probable that the longer the distance, the larger the corresponding RD cost. Therefore, a *cost checking* method is applied to early terminate mode decision if the RD cost (denoted as C_{cur}) of the current mode being checked is larger than $\gamma \cdot C_{\min}$, where γ is a regulating parameter and $C_{\min}$ is the minimum RD cost found so far. From exhaustive experimental results, it can achieve a good trade-off between encoding complexity reduction and video quality degradation by setting $\gamma = 1.0$.

TABLE VI
Simulation Environments

OS	Microsoft Windows XP
CPU	Intel E8500 3.16GHz
Memory	3.25GB
Profile	High
IntraPeriod	10
IDRPeriod	0
NumberReferenceFrames	5
RDOptimization	Low Complexity Mode
SubPelME	Enabled
SearchMode	UMHexagonS
SymbolMode	CABAC

Finally, based on the above analysis, the overall fast mode decision algorithm is presented as follows.

1) *Step 1:* Set the initial spatial weight w_S^0 for the current MB with (17) if the current frame is the first frame. Then if the current frame is an INTRA frame (I-frame), go to Step 7; else go to Step 2.
2) *Step 2:* Deduce the optimal central point P^* based on (8), and then build up the priority-based mode candidate list in an ascending order of distances between all INTER mode points and P^*. Go to Step 3.
3) *Step 3:* If the current frame is the first INTER frame (the first P-frame), check all INTER modes without *distance checking* and *cost checking*, and go to Step 6; else go to Step 4.
4) *Step 4:* Perform *distance checking* and then obtain SVIM. Initialize $C_{\min}$ as the maximum int value (e.g., $C_{\min} = 2^{31} - 1$ in 32-bit computers) and go to Step 5.
5) *Step 5:* Examine INTER modes in SVIM one by one. If the current mode is a sub-MB mode and the sub-MB mode decision hasn't been done, perform the sub-MB mode decision with all sub-MB modes in SVIM. If the current mode is a DIRECT mode (DIRECT16 × 16 or DIRECT8 × 8) and DIRECT modes have not been checked, check these two DIRECT modes. For the current mode (with its RD cost denoted as C_{cur}), if cost checking condition $C_{\text{cur}} > \gamma \cdot C_{\min}$ holds, then the INTER mode decision is early terminated, and go to Step 6; else if $C_{\min} > C_{\text{cur}}$, update $C_{\min} = C_{\text{cur}}$, check the next mode in SVIM. If all modes in SVIM have been checked, go to Step 6.
6) *Step 6:* Select the best INTER mode from all the INTER modes that have been checked. Update the spatial weight with (15). Go to Step 7.
7) *Step 7:* Perform INTRA mode decision and determine the best mode. Go back to Step 1 to process the next MB.

IV. Experimental Results

In order to examine the efficiency of the proposed algorithm, the H.264/AVC reference software JM13.2 [8] is employed for experiments. The results of five QCIF sequences (*Foreman*, *Grandma*, *Mother & Daughter*, *News*, and *Salesman*), five CIF sequences (*Coastguard*, *Mobile*, *Silent*, *Stefan*, and *Table*

TABLE VII

SUMMARY OF ENCODING RESULTS

Video	Wang2007			Liu2009			Zhao2008			Proposed		
	TS	ΔPSNR	ΔBR	TS	ΔPSNR	ΔBR	TS	ΔPSNR	ΔBR	TS	ΔPSNR	ΔBR
Foreman	40.22	−0.204	2.31	29.46	−0.079	4.34	46.31	−0.123	1.29	58.15	−0.095	0.81
Grandma	58.15	−0.054	−0.03	46.01	−0.112	5.08	54.75	0.001	0.89	62.99	−0.033	0.30
Mother	57.39	−0.113	0.20	43.46	−0.129	6.06	54.09	−0.046	0.72	62.48	−0.060	0.09
News	51.97	−0.093	2.15	41.42	−0.084	5.35	51.97	−0.045	1.73	61.29	−0.068	1.42
Salesman	49.51	−0.054	0.92	39.77	−0.128	5.09	52.89	0.004	1.37	61.19	−0.013	1.50
Coastguard	36.81	−0.093	2.17	28.80	−0.037	1.18	48.73	−0.081	0.45	59.79	−0.046	0.41
Mobile	31.64	−0.061	2.08	34.33	−0.062	2.08	45.10	−0.092	0.88	57.80	−0.068	1.22
Silent	53.00	−0.096	2.78	39.70	−0.145	2.83	51.24	−0.000	1.31	58.95	0.017	1.54
Stefan	32.36	−0.119	5.10	27.18	−0.044	1.31	42.10	−0.086	1.18	51.58	−0.042	1.79
Table	49.48	−0.117	3.47	38.14	−0.058	3.40	49.75	−0.051	1.12	57.99	−0.033	0.90
Parkrun	33.73	−0.047	−0.12	23.69	−0.024	0.07	44.55	−0.047	0.64	55.70	−0.021	0.17
Stockholm	56.53	−0.092	3.28	38.87	−0.046	3.52	55.36	−0.040	1.66	66.13	−0.027	−0.45
IPPP	48.10	−0.077	1.23	37.25	−0.079	4.58	48.97	−0.007	1.36	56.80	−0.018	1.14
IBBP	43.69	−0.112	2.83	34.55	−0.079	2.14	50.84	−0.094	0.85	62.21	−0.063	0.48
$Q_p = 24$	30.51	−0.050	1.62	27.55	−0.048	2.36	41.59	−0.062	1.18	53.05	−0.026	1.55
$Q_p = 28$	37.15	−0.058	1.56	30.73	−0.063	3.12	45.89	−0.056	1.22	57.09	−0.038	1.17
$Q_p = 32$	45.42	−0.077	1.82	35.64	−0.062	3.88	51.04	−0.053	1.31	60.91	−0.043	0.93
$Q_p = 36$	53.96	−0.110	2.19	40.55	−0.092	4.15	54.37	−0.046	1.14	62.70	−0.047	0.52
$Q_p = 40$	62.44	−0.181	2.95	45.05	−0.130	3.29	56.64	−0.036	0.68	63.78	−0.049	−0.12
Average	**45.90**	**−0.095**	**2.03**	**35.90**	**−0.079**	**3.36**	**49.91**	**−0.050**	**1.10**	**59.50**	**−0.041**	**0.81**

TABLE VIII

COMPARISON OF THE WORST PERFORMANCES

Algorithm	Video	GOP	Q_p	TS	ΔPSNR	ΔBR
Worst TS						
Wang2007	*Parkrun*	IPPP	24	18.04	−0.038	−0.30
Liu2009	*Parkrun*	IBBP	24	12.35	−0.014	−0.42
Zhao2008	*Parkrun*	IPPP	24	23.52	0.006	0.30
Proposed	*Parkrun*	IPPP	24	34.43	0.005	0.38
Worst ΔPSNR						
Wang2007	*Foreman*	IBBP	40	58.06	−0.500	2.63
Liu2009	*Mother*	IPPP	40	54.54	−0.278	5.93
Zhao2008	*Foreman*	IBBP	36	53.04	−0.201	−0.34
Proposed	*Foreman*	IBBP	36	63.69	−0.154	−1.48
Worst ΔBR						
Wang2007	*Stockholm*	IBBP	40	72.31	−0.250	12.89
Liu2009	*Mother*	IPPP	32	43.54	−0.099	10.22
Zhao2008	*Stockholm*	IBBP	32	59.26	−0.084	4.08
Proposed	*Salesman*	IBBP	24	62.78	−0.068	3.91

Tennis), and two SD sequences (*Parkrun* and *Stockholm*) are presented with two GOP structures (IPPP and IBBP). Due to lack of space, in the following tables, we use *Mother* and *Table* instead of *Mother & Daughter* and *Table Tennis* for short, respectively. In order to examine the performance at different bitrates, five Q_p values (Q_p = 24, 28, 32, 36, and 40) are tested. The simulation environments are given in Table VI. Especially, for the IPPP structure, there are 300 frames coded for each sequence; for IBBP GOP structure, there are 298 frames coded for each sequence. The search range is set to be 32 and 64 for QCIF/CIF sequences and SD sequences, respectively. The other parameters are set as defaults of the reference software.

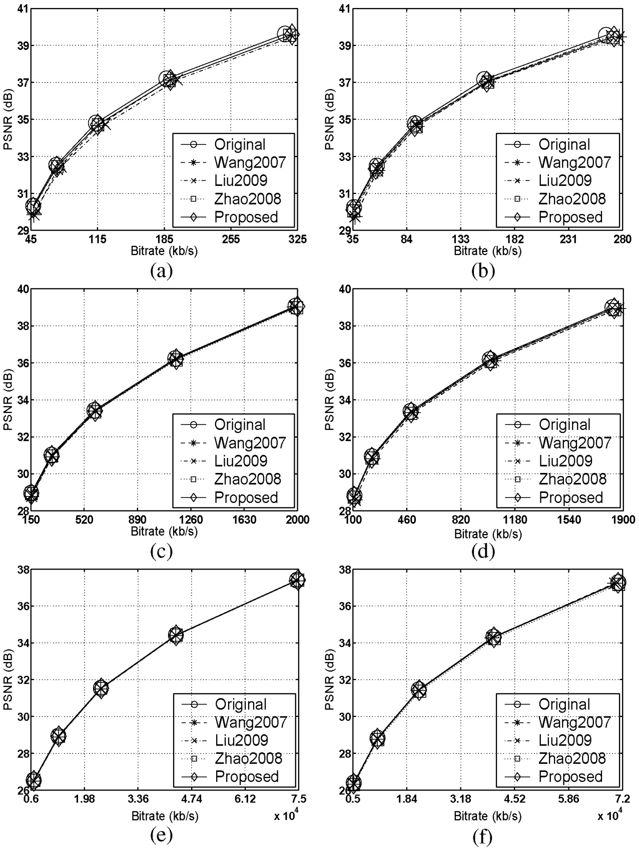
For comparison, three recent fast mode decision algorithms, including Wang2007 [10], Liu2009 [13], and our previous work Zhao2008 [16] are implemented. The performance is evaluated using the entire encoding time saving (TS, %), the peak-signal-to-noise-ratio decrease (ΔPSNR, dB) and the bitrate increase (ΔBR, %), where ΔPSNR and ΔBR are presented in (4) and (5), respectively; and TS is defined as

$$\mathrm{TS} = \frac{T_o - T_p}{T_o} \times 100\% \tag{18}$$

where T_o and T_p are the entire encoding time of the original encoder and the test algorithm (such as Wang2007, Liu2009, Zhao2008 and the proposed algorithm), respectively.

Comparative results are summarized in Table VII, where it can be observed that the proposed algorithm can achieve significantly better results than the other three algorithms in reducing the entire encoding time. In particular, the proposed algorithm works well for not only slow-motion or simple-context video sequences (such as *News* and *Silent*) but also fast-motion or complex-texture video sequences (such as *Mobile* and *Stefan*). Even for the SD sequences, the proposed algorithm could also get obviously better performance than the other three algorithms. The TS results for different Q_p values indicate that the proposed algorithm is robust for encoder complexity reduction even under small Q_p conditions, e.g., TS = 53.05% when Q_p = 24. Along with increase in Q_p, the TS performance can be further improved by the proposed algorithm.

Another observation on Table VII is that the proposed algorithm obtains less ΔPSNR and ΔBR than the other three algorithms. To illustrate the RD performance intuitively, the RD curves of QCIF (*Foreman*), CIF (*Coastguard*), and SD (*Parkrun*) sequences with both IPPP and IBBP structures are given in Fig. 3, where it can be observed that the proposed algorithm achieves almost the same RD performance as the original encoder.



(a) (b)

(c) (d)

(e) (f)

Fig. 3. RD curves. (a) *Foreman* (QCIF, IPPP). (b) *Foreman* (QCIF, IBBP). (c) *Coastguard* (CIF, IPPP). (d) *Coastguard* (CIF, IBBP). (e) *Parkrun* (SD, IPPP). (f) *Parkrun* (SD, IBBP).

For further studying the robustness of test algorithms, the worst encoding results for each comparative algorithm are given in Table VIII, where the worst TS, the worst ΔPSNR and the worst ΔBR are presented under a corresponding coding scenario. From Table VIII, it is observed that the proposed algorithm is more robust than the other three algorithms, i.e., the worst performance of the proposed algorithm is still significantly better than those of the other three algorithms, which is because spatial–temporal weight adaptation could make the proposed algorithm more suitable for scene changes or complex backgrounds.

V. Conclusion

An efficient fast mode decision algorithm was proposed based on mode adaptation in this paper. In particular, by utilizing the coding information of spatial–temporal neighboring blocks, a priority-based mode candidate list was constructed for effectively selecting the best coding mode. The H.264/AVC reference software JM13.2 was employed to evaluate the proposed algorithm. The comparative experimental results demonstrate that the proposed algorithm can greatly reduce the computational complexity of H.264/AVC encoding while maintaining almost the same RD performance as the original encoder, and can achieve significantly better results than the other three algorithms [10], [13] and [16].

References

[1] *Advanced Video Coding For Generic Audiovisual Services*, ISO/IEC 14496-10:2005(E) ITU-T Rec. H.264(E), Mar. 2005.

[2] Y. Huang, B. Hsieh, S. Chien, S. Ma, and L. Chen, "Analysis and complexity reduction of multiple reference frames motion estimation in H.264/AVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 4, pp. 507–522, Apr. 2006.

[3] P. Yin, H. C. Tourapis, A. M. Tourapis, and J. Boyce, "Fast mode decision and motion estimation for JVT/H.264," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, vol. 3. Sep. 2003, pp. 853–856.

[4] J. Lee and B. W. Jeon, "Fast mode decision for H.264 with variable motion block size," in *Proc. Int. Symp. Comput. Inform. Syst. (ISCIS)*, LNCS 2869. Nov. 2003, pp. 723–730.

[5] X. Jing and L. P. Chau, "Fast approach for H.264 INTER mode decision," *Electron. Lett.*, vol. 40, no. 17, pp. 1051–1052, Aug. 2004.

[6] I. Choi, J. Lee, and B. Jeon, "Fast coding mode selection with rate-distortion optimization for MPEG-4 part-10 AVC/H.264," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 12, pp. 1557–1561, Dec. 2006.

[7] D. Wu, F. Pan, K. P. Lim, S. Wu, Z. G. Li, X. Lin, S. Rahardja, and C. C. Ko, "Fast intermode decision in H.264/AVC video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 7, pp. 953–958, Jul. 2005.

[8] *H.264/AVC Reference Softwares* [Online]. Available: http://iphome.hhi.de/suehring/tml/

[9] Y. H. Kim, J. W. Yoo, S. W. Lee, J. Shin, J. Paik, and H.-K. Jung, "Adaptive mode decision for H.264 encoder," *Electron. Lett.*, vol. 40, no. 19, pp. 1172–1173, Sep. 2004.

[10] H. Wang, S. Kwong, and C.-W. Kok, "An efficient mode decision algorithm for H.264/AVC encoding optimization," *IEEE Trans. Multimedia*, vol. 9, no. 4, pp. 882–888, Jun. 2007.

[11] H. Wang, S. Kwong, and C.-W. Kok, "Efficient prediction algorithm of integer DCT coefficients for H.264/AVC optimization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 4, pp. 547–552, Apr. 2006.

[12] A. C. W. Yu, G. R. Martin, and P. Heechan, "Fast inter-mode selection in the H.264/AVC standard using a hierarchical decision process," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 2, pp. 186–195, Feb. 2008.

[13] Z. Liu, L. Shen, and Z. Zhang, "An efficient intermode decision algorithm based on motion homogeneity for H.264/AVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 1, pp. 128–132, Jan. 2009.

[14] S. H. Ri, Y. Vatis, and J. Ostermann, "Fast inter-mode decision in an H.264/AVC encoder using mode and lagrangian cost correlation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 2, pp. 302–306, Feb. 2009.

[15] H. Zeng, C. Cai, and K.-K. Ma, "Fast mode decision for H.264/AVC based on macroblock motion activity," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 4, pp. 491–499, Apr. 2009.

[16] T. Zhao, H. Wang, X. Ji, and S. Kwong, "2-D map-based fast mode decision for H.264/AVC," in *Proc. Pacific-Rim Conf. Multimedia (PCM)*, LNCS 5353, Dec. 2008, pp. 148–157.

[17] T. M. Chan, K. F. Man, S. Kwong, and K. S. Tang, "A jumping gene paradigm for evolutionary multiobjective optimization," *IEEE Trans. Evol. Comput.*, vol. 12, no. 2, pp. 143–159, Apr. 2008.

Tiesong Zhao (S'08) received the B.S. degree in electrical engineering from the University of Science and Technology of China, Hefei, China, in 2006. He is currently pursuing the Ph.D. degree from the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong.

His current research interests include video-object segmentation and algorithm optimization of video coding standards.

Hanli Wang (M'08) received the B.S. and M.S. degrees in electrical engineering from Zhejiang University, Hangzhou, China, in 2001 and 2004, respectively, and the Ph.D. degree in computer science from City University of Hong Kong (CityU), Kowloon, Hong Kong, in 2007.

From 2007 to 2008, he was a Research Fellow with the Department of Computer Science, CityU. From 2007 to 2008, he also was a Visiting Scholar with Stanford University, Palo Alto, CA, invited by Prof. C. K. Chui. From 2008 to 2009, he was a Research

Engineer with Precoad, Inc., Menlo Park, CA. From 2009 to 2010, he was an Alexander von Humboldt Research Fellow in University of Hagen, Hagen, Germany. In 2010, he joined the Department of Computer Science, Tongji University, Shanghai, China, as a Professor. His current research interests include digital video coding, image processing, pattern recognition, and video analysis.

Sam Kwong (SM'04) received the B.S. and M.S. degrees in electrical engineering from the State University of New York, Buffalo, and the University of Waterloo, Waterloo, Ontario, Canada, in 1983 and 1985, respectively, and received the Ph.D. degree from the University of Hagen, Hagen, Germany, in 1996.

From 1985 to 1987, he was a Diagnostic Engineer with Control Data Canada, Mississauga, Toronto, Canada. He later joined Bell Northern Research Canada, Ottawa, Canada as a Scientific Staff Member. In 1990, he joined the Department of Electronic Engineering, City University of Hong Kong, Kowloon, Hong Kong, as a Lecturer, and is currently an Associate Professor with the Department of Computer Science at the same university. His current research interests include video coding, evolutionary algorithms, speech processing and recognition, and digital watermarking.

C.-C. Jay Kuo (F'99) received the B.S. degree in electrical engineering from the National Taiwan University, Taipei, Taiwan, 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 and a Professor of electrical engineering, computer science and mathematics with the Department of Electrical Engineering and Integrated Media Systems Center, University of Southern California, Los Angeles. His current research interests include digital image/video analysis and modeling, multimedia data compression, communication and networking, and biological signal/image processing. He is the Co-author of about 170 journal papers, 800 conference papers, and 10 books.

Dr. Kuo is a Fellow of the International Society for Optical Engineers.