An Effective Combination of Power Scaling for H.264/AVC Compression

Hyun Kim, Chae Eun Rhee, and Hyuk-Jae Lee

Abstract—This brief proposes a novel method to determine the best combination of operation conditions for multiple power-scaling schemes. The power saving and rate-distortion performances of individual schemes are simulated, and then, the combined effects are modeled to obtain the best operation combination. The optimized combinations are defined as a power-level table. The proposed power-aware design is tested with four popular power-saving schemes and simulations show that a power saving of ~25% is achieved at the sacrifice of <0.172 dB Bjontegaard Delta peak signal-to-noise ratio degradation.

Index Terms—H.264/AVC, power reduction, power-aware design, power-scaling scheme, real-time application.

I. INTRODUCTION

Extensive research has been undertaken to reduce power consumption by an H.264 encoder given the increasing use of mobile devices with limited battery capacity. In [1], the concept of a power-aware design is introduced and various power-scaling schemes in H.264 encoders are presented. In [2], the power consumption is controlled to meet the predefined target and various operation conditions for integer motion estimation (IME), fractional motion estimation (FME), and intraprediction (IP) are determined. In [3], the operation conditions are controlled based on video complexity and the remaining battery capacity. In [4] and [5], power-rate-distortion (P-R-D) models are proposed in which the relationships between power, rate, and distortion are derived. The P-R-D models depend on the video's characteristics, which are unknown at run time. In addition, a large number of simulations are needed when the video sequence is changed or when a new scheme is added because all combinations of schemes should be simulated in each video sequence. Because of these limitations, the P-R-D models cannot be used in practical use to determine the best operation condition.

This brief proposes a novel power-aware design for an H.264/advanced video coding (AVC) encoder; a design that optimizes rate-distortion (R-D) performance. The proposed power-aware design changes the operation condition in the encoder using various power-scaling schemes. To develop an effective combination of various power-scaling schemes, this brief formulates the model to estimate the cross-effects on power consumption among the scaling schemes. Using the proposed modeling approach, the number of simulations required to determine

H. Kim and H.-J. Lee are with the Inter-University Semiconductor Research Center, Department of Electrical Engineering, Seoul National University, Seoul 151-742, Korea (e-mail: snusbkh0@capp.snu.ac.kr; hyuk_jae_lee@capp.snu.ac.kr).

C. E. Rhee is with the School of Information and Communication Engineering, Inha University, Incheon 402-751, Korea (e-mail: chae.rhee@inha.ac.kr).
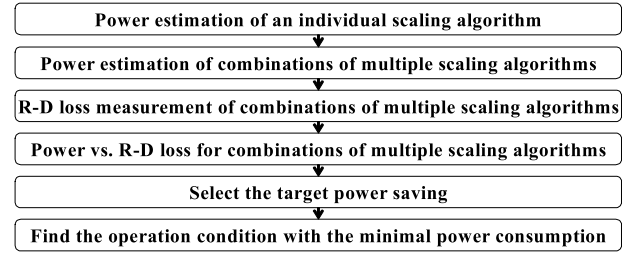
Fig. 1. Generation of a power-level table.

the best combination of power-scaling schemes is markedly reduced. Consequently, the set of power-scaling schemes is easily composed. Based on the estimated power saving and simulated R-D performance, a power-level table is defined. Each power level in that table has corresponding operation conditions that minimize the R-D loss for a given power consumption target. To achieve better R-D performance, four different power-level tables are defined depending on video size and motion speed and the most proper power level is selected adaptively. Simulation results show that the average Bjontegaard Delta peak signal-to-noise ratio (PSNR) [6] degradation for achieving power savings of 25% is <0.172 dB.

II. POWER-AWARE DESIGN

The proposed power-aware design aims to control power consumption to meet a power budget target by combining various individual power-saving algorithms that are widely used for H.264/AVC encoders.

A. Generation of a Power-Level Table

The power-level table defines a list of operation conditions for various power-saving algorithms along with their power consumptions. The operation conditions are chosen to achieve the best video quality for the given power consumption level as discussed.

Generation of a power-level table includes six steps as shown in Fig. 1. In the first step, the power saving from application of an individual power-reduction algorithm is estimated by simulation. For each algorithm, simulation is performed under various operation conditions and the power consumption for each of the conditions is estimated. Such simulations are performed for every power-saving algorithm to be used for power control. For accurate power estimation, a postlayout simulation for each algorithm is desirable.

In the second step, the power savings obtained by applying various combinations of power-scaling algorithms are estimated. Estimation of power consumptions for all possible combinations of operation conditions is very time consuming. To save time, a power consumption model is used to derive the power consumptions associated with the various power-scaling algorithm combinations instead of using postlayout simulation. That model combines the power simulation results from the first step and enables estimation of power consumption of the various power-scaling algorithm combinations. Details of the power consumption model are presented in Section III.

In the third step, the corresponding R-D loss is measured for the various algorithm combinations estimated in the second step. This measurement is obtained by software simulation, which takes far less time than postlayout simulation, making it possible to obtain R-D losses for all possible combinations. From the power saving and R-D loss obtained in the second and third steps, respectively,

the relationship between power saving and corresponding R-D loss is obtained for all possible combinations of power-saving algorithms in the fourth step. The fifth step selects a fixed number of power-saving targets. The final step finds the operating conditions that minimize R-D loss.

The proposed power-aware design classifies the input video into four categories based on video size and motion characteristics. A power table is generated separately for four categories: 1) fast-large; 2) fast-small; 3) slow-large; and 4) slow-small videos. A large video sequence has a video width >1000 pixels. A video sequence is classified as slow or fast motion if the average magnitude of the motion vectors in the previous 30 frames is either smaller or larger, respectively, than a predefined threshold, one pixel, obtained by experiments.

B. Power-Scaling Algorithm

The encoding period is divided into small intervals and the power consumption is controlled independently for each period. Suppose that the total power budget (P_{TOTAL}) is given for an encoder to operate for a certain period of time. Then, let P_{CUR} denotes the available power budget for the current period, P_{PAST} denotes the power consumption used in the preceding periods, and P_{FUR} denotes the estimated power budget for the future. The exact value of P_{FUR} cannot be evaluated so that it is estimated by allocating an average value, which is calculated by dividing P_{TOTAL} by the expected number of periods, to each future period. The current power budget P_{CUR} is then calculated using

$$P_{CUR} = P_{TOTAL} - (P_{PAST} + P_{FUR}). \qquad (1)$$

The flow of the power-scaling algorithm is composed with four steps. The first step is to calculate the power budget target for P_{CUR} using (1). The second step selects the appropriate power-level table among the four size/motion-based power-level tables for the input video. The third step selects the appropriate power level from the table selected in step two that will meet the P_{CUR} target. After encoding the current period with the chosen level, the fourth step identifies the video characteristics of the current period and determines the amount of power consumed. Those four steps are repeated until the encoding of all periods is completed. In this brief, the selected power-level update period is 1 min. In general, a shorter update period may be more effective because the change of video characteristics can be reflected promptly, whereas the computational complexity of the update overhead is very small. Simulation results show that the effect of the update period on the coding efficiency is small so that 1 min is chosen.

III. POWER ESTIMATION MODEL

Power-scaling algorithms are classified into two types: 1) one applicable for interframe prediction; and 2) the other suitable for intraframe prediction. By applying those two types, total power saving (PS_{TOTAL}) is obtained using

$$PS_{TOTAL} = PS_{P-frame} \times (1 - 1/P) + PS_{I-frame} \times 1/P \qquad (2)$$

where $PS_{P-frame}$ and $PS_{I-frame}$ denote the power savings during interframe (P-frame) and intraframe (I-frame) operations, respectively, and where P represents the period of the frames encoded as an I-frame. Of the two power saving terms in (2), $PS_{P-frame}$ is analyzed first. There are five main hardware-based operations within interframe prediction: 1) IME; 2) FME; 3) IP; 4) adaptive deblocking filter (ADF); and 5) variable length coding (VLC). The total power saving is a summation of the power savings from these five hardware modules. A number of power-scaling algorithms are proposed for IME, FME, and IP. In contrast, scaling power consumption by

ADF and VLC is not reported extensively. Furthermore, the amount of power computation by VLC or ADF is relatively small and the option to turn-OFF ADF may significantly degrade the subjective image quality, such as existence of blocking artifacts. Thus, power scaling by ADF or VLC is not considered hereafter in this brief.

The power consumption of a P-frame, $PS_{P-frame}$, is the summation of PS_{IME}, PS_{FME}, and PS_{IP} which denote the power savings in IME, FME, and IP operations, respectively

$$PS_{P-frame} = PS_{IME} + PS_{FME} + PS_{IP}. \qquad (3)$$

Power-scaling algorithms for IME, FME, and IP can be classified into two categories. In the first category, the computational complexity of the operation is controlled. A search algorithm for IME is an example of that category. In the second category, the execution frequency of the operation is controlled. For example, an early SKIP mode detection can result in the elimination of motion estimation operations. Then, each of three power savings (PS_{IME}, PS_{FME}, and PS_{IP}) can be formulated as a summation of the power savings in the two categories. For example, the power saving of IME is formulated as follows:

$$PS_{IME} = PS_{IME,RC} + PS_{IME,RF} \qquad (4)$$

where $PS_{IME,RC}$ and $PS_{IME,RF}$ denote the power savings achieved by the algorithms in the first and second categories, respectively. For the estimation of $PS_{IME,RC}$, a postlayout simulation may be necessary to obtain an accurate result. On the other hand, the estimation of $PS_{IME,RF}$ may be obtained without postlayout simulation. Instead, a software simulation can be used to obtain the frequency of the excluded executions. Let PC_{IME} denotes the power consumption of IME without any power saving scheme and let $FE_{IME,RF}$ denotes the frequency of the operations that are excluded by the power saving in the second category. Then, the power saving by the secondary algorithm is $FE_{IME,RF} \times PC_{IME}$ and consequently (4) becomes

$$PS_{IME} = PS_{IME,RC} + [FE_{IME,RF} \times PC_{IME}]. \qquad (5)$$

The power saving associated with FME and IP operations are obtained similar to (5), and consequently, total power saving for interframe prediction is formulated as follows:

$$\begin{aligned} PS_{P-frame} = & \; PS_{IME,RC} + [FE_{IME,RF} \times PC_{IME}] \\ & + PS_{FME,RC} + [FE_{FME,RF} \times PC_{FME}] \\ & + PS_{IP,RC} + [FE_{IP,RF} \times PC_{IP}] \qquad (6) \end{aligned}$$

where $FE_{FME,RF}$ and $FE_{IP,RF}$ denote the frequencies of the reduced FME and IP operations, respectively.

The estimation of power consumption by (6) needs the estimation of $PS_{IME,RC}$, $FE_{IME,RF}$, PC_{IME}, $PS_{FME,RC}$, $FE_{FME,RF}$, PC_{FME}, and $PS_{IP,RC}$, $FE_{IP,RF}$, PC_{IP}. Note that $PS_{IME,RC}$, $PS_{FME,RC}$, and $PS_{IP,RC}$ are the power savings of individual schemes so that just a single postlayout simulation is necessary to obtain these values. $FE_{IME,RF}$, $FE_{FME,RF}$, and $FE_{IP,RF}$ are the parameters that can be obtained from software simulation which is much faster than hardware postlayout simulation. PC_{IME}, PC_{FME}, and PC_{IP} can also be obtained by a single postlayout simulation. Therefore, the number of postlayout simulation depends on the number of power scaling schemes, but does not depend on the number of all possible combinations of available power scaling schemes. In summary, the proposed power consumption model reduces the number of required postlayout simulations from $O(N!)$ to $O(N)$ complexity, where N is the number of available power-scaling algorithms.

To derive $PS_{I-frame}$, the second term in (2), the power saving in the intraframe prediction is analyzed. Note that intraframe prediction

TABLE I
EFFECT OF THE POWER-SAVING ALGORITHM

Algorithm	Category	Affected modules
FME prediction module reduction	1	FME
IME search range control	1	IME
Early skip mode decision	2	FME, IP
Intra-frame period control	2	IME, FME

requires less power consumption than that used by interframe prediction. Thus, intraframe prediction inherently achieves power saving of amount $(PC_{IME} + PC_{FME})$ because IME and FME are not performed

$$PS_{I-frame} = PC_{IME} + PC_{FME}. \tag{7}$$

IV. EXAMPLE OF POWER-AWARE DESIGN WITH FOUR POWER-SCALING ALGORITHMS

In this section, four power-scaling algorithms are used as examples to explain the proposed power-aware design.

A. Four Power-Scaling Algorithms

1) FME Prediction Mode Reduction: The prediction mode reduction for FME decides the number of FME modes to be performed based on IME results [7]. A decreased FME complexity leads to the reduction of the power consumption.

2) IME Search Range Control: This search range control scheme adjusts the amount of IME computation. In this brief, a content-adaptive search range [8] is used and the search range is scaled down by various ratios.

3) Early SKIP Mode Decision: Among various early SKIP mode decision algorithms, the algorithm presented in [9] determines the SKIP mode after the IME stage. If the current macroblock is determined to be the SKIP mode, the following FME and IP operations are omitted and the power consumption required for FME and IP is reduced.

4) Intraframe Period Control: Power consumption is reduced as the intraframe period is decreased.

By applying the above four algorithms from (2) through (7), an example power consumption model is derived. $PS_{P-frame}$ is analyzed first. For the derivation of PS_{IME} in (5), the IME search range control is the only applicable algorithm. Note that the IME search range control is of the first category. Therefore, only the first term in (5) is effective, whereas $FE_{IMERF} = 0$. For the derivation of PS_{FME}, both the FME mode reduction and the early SKIP schemes affect the power saving. The FME mode reduction is a first-category algorithm. The early SKIP is a second-category algorithm which affects the power consumption of FME and IP. The intraframe period control can be considered as the second category, but it is related to the $PS_{I-frame}$. Thus, (7) is used for the estimation. Table I summarizes the effect of four algorithms for power estimation.

B. Power Simulation of Individual Algorithms

The amount of power saving and the corresponding R-D loss associated with application of the four power-scaling algorithms are obtained independently by simulation. For the power simulation, the hardware-based H.264 encoder [10] is synthesized using a Synopsys Design Compiler with a $0.13-\mu m$ library and the power consumption is measured with postlayout simulation. The R-D performance is obtained by software simulation of the hardware reference model, which gives exactly the same result as that from the hardware-based encoder. For derivation of the R-D performance, 12 video sequences are used: 1) three slow-motion common
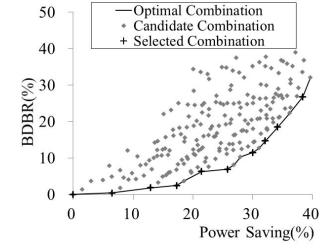

Fig. 2. Power saving versus BDBR change for combinations of power-scaling schemes for HD fast-motion videos.

intermediate format (CIF) videos (*Container*, *News*, and *Sean*); 2) three fast-motion CIF videos (*Table*, *Bus*, and *Stefan*); 3) three slow-motion hard-drawn (HD) videos (*Aspen*, *Sunflower*, and *Intotree*); and 4) three fast-motion HD videos (*Factory*, *Pedestrian area*, and *Tractor*). The number of frames in each video is 100, while the quantization parameter (QP) values are 20, 24, 28, and 32. The encoding configuration uses a base profile and the group of pictures structure is IPPP.... The operating clock frequency is 50 MHz for CIF videos and 166 MHz for HD videos to obtain a frame rate of 30 frames/s for the hardware-based H.264 encoder [10].

C. Estimation of the Combined Power Saving and Derivation of the Optimal Operating Conditions

This section describes the generation of a power-level table based on the results of the four power-scaling algorithms described in Fig. 1. The relationships between Bjontegaard Delta bitrate (BDBR) [6] change and power saving obtained from various combinations of available algorithm operation conditions are plotted in Fig. 2. Note that the BDBR change can be obtained from software simulation. The horizontal axis represents the power saving, whereas the vertical axis does the BDBR change. Each point in Fig. 2 represents the BDBR change and power saving derived from a given operation condition. In Fig. 2, the points at the lower-right portion of the plots represent better power saving than those at the upper-left because the lower-right points have small BDBR increases at the same or similar power saving levels. Among the gray points in Fig. 2, the points that have the smallest BDBR change relative to the obtained power savings are connected with the segmented line. The points along the segmented line represent the operation conditions providing the maximum power saving for a given BDBR change. Note that only the results for HD fast-motion videos are shown in Fig. 2 because the results for the other types of videos are similar to Fig. 2.

D. Generation of a Power Table

The final step, in this example, of the power-aware design is the generation of power levels that are associated with the optimal operating conditions of the four algorithms. To this end, the 10 operating conditions are selected from those shown in Fig. 2 (marked with + in Fig. 2). From these operating conditions, power-level tables are developed as shown in Table II. At power level 0, none of the four power-scaling schemes is applied. Level 9 offers the largest amount of power saving. The prediction mode reduction (PMR), search range (SR) control, early skip (ES) mode decision, and IP columns represent the operation conditions of the prediction mode reduction, search range control, early SKIP mode decision and intraframe period control, respectively. In this table, the number of the power levels is chosen as 10 based on simulation results showing that the improvement of coding efficiency by increasing the number of levels is saturated around the number of 10.

For PMR, three power reduction modes are used. In Mode 5, two FME operations are selected from the 16×16, 16×8, and 8×16

TABLE II
POWER-LEVEL TABLE

Level	CIF Slow-Motion					HD Slow-Motion					CIF Fast-Motion					HD Fast-Motion				
	P MR	SR	ES	IP	PC(mW)/PS(%)	P MR	SR	ES	IP	PC(mW)/PS(%)	P MR	SR	ES	IP	PC(mW)/PS(%)	P MR	SR	ES	IP	PC(mW)/PS(%)
0	-	-	-	-	17.9/0	-	-	-	-	81.1/0.0	-	-	-	-	20.2/0.0	-	-	-	-	90.1/0.0
1	-	-	O	-	15.4/14.0	5	-	-	-	75.7/6.7	5	-	-	-	18.8/6.7	5	-	-	-	84.3/6.5
2	-	1/2	O	-	14.2/20.4	5	1/2	-	-	70.1/13.6	5	-	O	-	18.0/10.8	3	-	-	-	78.4/12.9
3	-	1/4	O	-	13.7/23.6	5	1/4	-	-	67.3/17.0	5	1/2	-	-	17.4/13.9	1	-	-	-	74.6/17.3
4	5	1/4	O	-	12.9/27.7	3	1/2	-	-	64.6/20.3	5	1/2	O	-	16.5/18.1	3	1/2	-	-	70.8/21.4
5	5	1/6	O	-	12.7/28.8	3	1/4	-	-	61.9/23.7	3	1/2	O	-	15.4/23.8	1	1/2	-	-	66.9/25.8
6	3	1/6	O	-	12.0/32.9	1	1/4	-	-	58.2/28.2	1	1/2	O	-	14.6/27.5	1	1/4	-	-	62.8/30.3
7	1	1/6	O	-	11.5/35.6	1	1/4	O	-	54.8/32.4	1	1/2	O	30	14.2/29.6	1	1/4	-	30	62.2/32.1
8	1	1/9	O	-	11.4/36.3	1	1/6	O	30	52.4/35.3	1	1/6	O	60	13.5/33.3	1	1/4	-	15	59.4/34.1
9	1	1/9	O	60	11.1/38.1	1	1/6	O	15	51.0/37.1	1	1/6	O	10	12.5/38.0	1	1/4	O	10	55.6/38.3
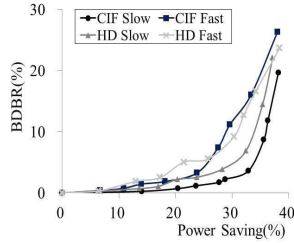
Fig. 3.   Power saving versus BDBR change performance of the power-level design.
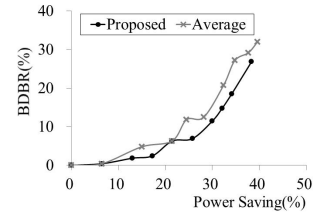

Fig. 4.   Comparison of results from the proposed power-levels method and from applying average power levels for HD-size fast-motion videos.

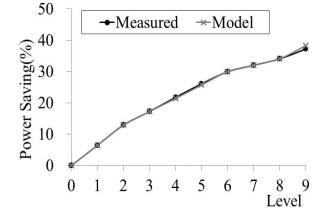Fig. 5.   Comparison of measured and modeled power-level saving results for HD-size fast-motion videos.

partitions and three are selected from four 8×8 partitions. In Mode 3, one FME operation is selected from 16×16, 16×8, and 8×16 partitions and two are selected from four 8×8 partitions as described in [7]. In Mode 1, only a single mode, determined to be the best mode in the IME operation, is selected for the FME operation. The ES algorithm is selected for use at lower power levels in CIF videos than those in HD videos, as shown in Table II, because the ES algorithm is more useful for low resolution videos. For SR, the search ranges are adjusted as the 1/2, 1/4, 1/6, or 1/9 of the original range in both horizontal and vertical directions. For IP, the intraframe period varies among 10, 15, 30, and 60 In the table, columns 6, 11, 16, and 21 show the power consumption in milliwatt as well as the relative power saving in comparison with the power consumption of level 0.

V. SIMULATION RESULTS

A. Performance Estimation of the Power-Aware Design

The performance of the power-aware design is assessed by performing simulations. In Fig. 3, the relationships between the BDBR changes and the power savings for the four video types are shown. The 20 test video sequences with each sequence consisting of 100 frames are encoded with four QP values (i.e., 20, 24, 28, and 32). The relationships indicate that an ~25% power saving can be obtained with a <5% BDBR increase on average. However, power savings of >25% are associated with a significant BDBR increase, particularly in fast-motion videos.

A further simulation evaluates the effectiveness of the four different power-level tables according to video size and motion characteristics. For that comparison, a new power table is generated from the power saving and BDBR values averaged over all video sequences used in Section IV-B, regardless of video size or motion characteristics. Fig. 4 compares the average power levels with the proposed power levels in Table II for HD fast-motion videos. Note that the power consumption without power saving scheme is 901 mW. Compared with the proposed power-level results, application of the average

power levels increases the BDBR by >86% when the power saving is 34%. The results for other video types are similar to Fig. 4.

Other simulations evaluate the accuracy of the power saving model in Section III. Fig. 5 shows the results only for the HD fast-motion videos. The black graph labeled measured shows the average power savings from the postlayout simulation of three frames of a test image and the gray graph labeled model shows the power savings from the power saving model. The measured and modeled results are nearly identical at most power levels. The results for other video types are similar to Fig. 5 and the maximum difference between the measured and proposed is 2.9% at power level 3 for the CIF fast-motion video.

A further simulation evaluates the effectiveness of adaptive controlling the power level. The simulation results from the proposed approach are compared with the results from a fixed power level. For both adaptive and fixed level approaches, the power saving target is 35%. To achieve this goal, the fixed level control uses power levels 9 and 8 for CIF slow-motion and fast-motion videos, respectively. For the simulations, three CIF-size fast-motion sequences and three CIF-size slow-motion sequences are used with each sequence comprising 1500 frames. Note that it is impossible to run postlayout simulation for 1500 frames. Therefore, a combination of software profiling and hardware simulation is performed for the power estimation of 1500 frames. The simulation results in Table III show that, in comparison with the fixed approach, BDBR decreases by an average of 8.44% when the adaptive approach is applied.

TABLE III
BDBR IMPROVEMENT BY THE ADAPTIVE LEVEL CONTROL

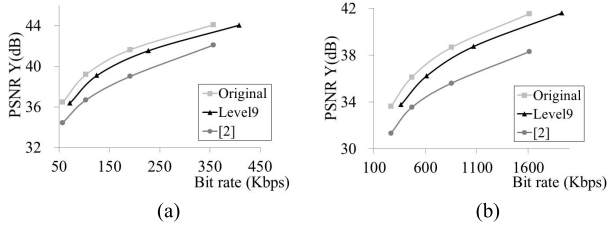| Sequence	Fast1	Fast2	Fast3	Slow1	Slow2	Slow3
BDBR (%)	-2.58	-16.59	-5.84	-5.71	-6.88	-13.03

Fig. 6. Comparison of the R-D performances with previous work. (a) *Akiyo*. (b) *Foreman*.

B. Comparison With a Previous Power-Aware Design

In this section, the proposed power-aware control system is compared with a previous power-aware design described in [2]. To compare the performance of the proposed power-aware design and the one in [2], comparison of the R-D performance is made under the same power consumption targets.

Fig. 6 shows the R-D performance when the power-saving target is a 40% reduction. In this brief and in [2], R-D curves for the slow-motion and fast-motion CIF-size *Akiyo* and *Foreman* sequences, respectively, are available. Thus, those videos are used for the comparison. In Fig. 6, the graph labeled original represents the R-D curve with no power-scaling scheme applied, the graph labeled level 9 represents the R-D results when the level 9 operating conditions for CIF slow-motion and fast-motion in Table II are applied, and the graph labeled [2] is the R-D curve from [2]. In Fig. 6(a), the PSNR difference is >2 dB at 200 kb/s. In Fig. 6(b), the PSNR difference is >2 dB at 1600 kb/s. Note, in Fig. 6, that the algorithm used in [2] results in significant R-D performance degradation at a power saving target reduction of 40%. In contrast, the proposed power-aware system combining four types of algorithms can reduce power consumption by ∼40% without marked R-D performance decrease. The main difference between the proposed design and that in [2] lies in that the proposed design includes various options to select IME complexity, whereas the design in [2] often gives up IME operation completely when the target power saving is aggressive such as 40%. Therefore, a large number of blocks are encoded as IP mode in the design in [2] for 40% saving which significantly degrades the R-D performance.

VI. CONCLUSION

The proposed power-aware design offers a large number of possible options among which the best operation condition is selected for a given target saving. This large number of possible options is possible because of the power consumption model of (2) that speeds up the estimation of power consumption for various operating conditions. It reduces the estimation time from $O(N!)$ to $O(N)$ complexity, where N is the number of available power-scaling algorithms. Without this power consumption model, it may take too much time to estimate the power consumption of all these various options. In latest smartphones, the encoding operation contributes to ∼25%–40% of the total power consumption for video capture applications. When 35% of power consumption for encoding is saved by the proposed algorithm, the total power saving of the smartphone may be ∼8.75%–14%.

REFERENCES

[1] C. J. Lian, P. C. Tseng, and L. G. Chen, "Low-power and power-aware video codec design: An overview," *Chin. Commun.*, vol. 3, no. 5, pp. 45–51, Oct. 2006.

[2] W.-C. Chang, G.-L. Li, and T.-S. Chang, "Power-aware coding for H.264/AVC video encoder," in *Proc. VLSI Design/CAD Symp.*, Aug. 2009.

[3] A. K. Kannur and B. Li, "Power-aware content-adaptive H.264 video encoding," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Apr. 2009, pp. 925–928.

[4] Z. He, W. Cheng, and X. Chen, "Energy minimization of portable video communication devices based on power-rate-distortion optimization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 5, pp. 596–608, May 2008.

[5] J. Kim, J. Kim, G. Kim, and C.-M. Kyung, "Power-rate-distortion modeling for energy minimization of portable video encoding devices," in *Proc. IEEE 54th Int. Midwest Symp. Circuits Syst. (MWSCAS)*, Aug. 2011, pp. 1–4.

[6] G. Bjontegaard, "Calculation of average PSNR differences between RD curves," in *Proc. VCEG-M33 ITU-T Q6/16*, Austin, TX, USA, Apr. 2001.

[7] T.-C. Chen, Y.-W. Huang, and L.-G. Chen, "Fully utilized and reusable architecture for fractional motion estimation of H.264/AVC," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 5. May 2004, pp. 9–12.

[8] J.-S. Jung, D.-U. Moon, and H.-J. Lee, "Computation reduction of H.264/AVC motion estimation by search range adjustment and partial cost evaluation," in *Proc. Int. Conf. Electron., Inf., Commun.*, Jun. 2008, pp. 229–233.

[9] H. Kim, C. E. Rhee, J.-S. Kim, S. Kim, and H.-J. Lee, "Power-aware design with various low-power algorithms for an H.264/AVC encoder," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2011, pp. 571–574.

[10] C. E. Rhee, J.-S. Jung, and H.-J. Lee, "A real-time H.264/AVC encoder with complexity-aware time allocation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 12, pp. 1848–1862, Dec. 2010.