Analysis of Power Consumption of H.264/AVC-based Video Sensor Networks through Modeling the Encoding Complexity and Bitrate

Bambang A.B. Sarif, Panos Nasiopoulos and Victor C.M. Leung

Department of Electrical and Computer Engineering,
University of British Columbia
Vancouver, Canada
{bambangs, panosn, vleung}@ece.ubc.ca

Mahsa T. Pourazad

Department of Electrical and Computer Engineering,
University of British Columbia
TELUS Communications Inc.
Vancouver, Canada
pourazad@ece.ubc.ca

Abstract—**The H.264/AVC video encoding standard has many advanced features that can be tailored to suit a wide range of applications. In order to obtain optimal coding performance in video sensor networks (VSNs), it is essential to find the right setting parameters for the encoder. There is a trade-off between required energy for encoding and transmission of video content in VSNs that can be exploited to minimize total power consumption. In this study, we model the complexity and bitrate in H.264/AVC codec. By using the model, the trade-off between encoding and transmission energy consumption is further exploited. Our experiments show that the complexity modeling error is less than or equal to 3.45%. However, the bitrate modeling error that we obtain is less than or equal to 11.6%.**

Keywords-H.264/AVC; complexity and bitrate modeling; energy consumption model; and video sensor network

I. INTRODUCTION

With the increasing concern about security in homes or public spaces, the demands for monitoring and surveillance systems is growing. In this regard, video sensor networks (VSNs) offer an alternative to several existing monitoring technologies [1], [2]. However, unlike the traditional sensor networks which require negligible power to process captured data in the sensor nodes, VSNs need significant processing power to encode and transmit the captured videos. With the limitations of energy resources in VSNs, maximizing the power efficiency of coding and transmission operations becomes very important. In general, there is a tradeoff between encoding complexity and compression performance in the sense that to obtain higher compression performance (i.e., lower bit rate), more complex and computationally expensive encoding scheme is required. On the other hand, transmission of lower bitrate content requires less amount of energy. Fig. 1 illustrates the relationship between coding complexity, compression performance and the required power for encoding and transmission of the content. It can be observed that, to minimize the overall VSN power consumption, encoding process needs to be handled carefully. Among the existing video coding standards, H.264/AVC is the most widely used standard in the consumer market [3], [4]. Some of the existing studies on the performance of H.264/AVC codec look into maximizing the coding performance without considering the total power
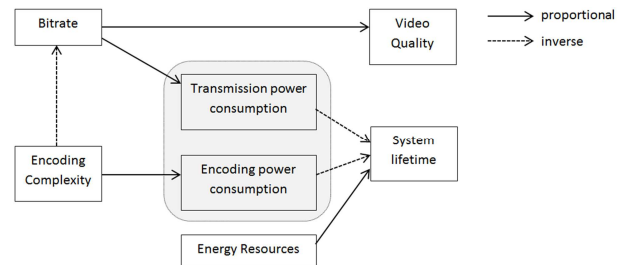

Figure 1. Relation between encoding and transmission power consumption

consumption of the coding process [3], [5]. J.J. Ahmad et al. [6] studied the required energy for encoding and transmission of video content in the case of using H.264/AVC codec. Unfortunately, the number of configuration settings considered for the encoder in that study is limited. To address this issue, we extended the study in [6] by including more encoder configuration settings in our previous work [7]. We proposed a guideline table for encoder configuration setting which include different combinations of coding complexity and coding efficiency in terms of bitrate that produces compressed videos with similar quality in terms of peak signal to noise ratio (PSNR). Our study shows that the energy consumption of a VSN can be reduced by carefully selecting the encoder settings at each VSN node based on the proposed table.

This paper is an extension to our previous work [7] where the relationship between coding complexity and coding efficiency (in terms of bitrate) of H.264/AVC codec is modeled. By using this model, the trade-off between encoder complexity and bitrate can be further elaborated, unrestrained with the encoder setting parameters. The rest of the paper is organized as follows. Section II describes the H.264/AVC encoding complexity and bitrate modeling. The encoding and transmission power consumption model is then discussed in Section III. Conclusions are drawn in Section IV.

II. H.264/AVC COMPLEXITY AND BITRATE MODELING

H.264/AVC is a block-based hybrid video coding standard utilizing intra-frame and inter-frame prediction. While inter-frame prediction is more involved than intra-frame prediction, it results in lower bitrate. By increasing the number of inter-frames coded picture within a successive video stream, i.e., group of picture (GOP) size, the bitrate of the coded video is reduced at the cost of higher encoding complexity. In the case of inter-frame prediction, the complexity and bitrate can be controlled by adjusting the search range (SR) in motion estimation process. The SR determines the size of searching area in the reference frame to find the best match to be used for inter prediction. Increasing the SR size may result in better compression performance at the cost of increased complexity. However this observation is quite content dependant and there are cases where increasing the value of SR does not provide significant benefit in terms of compression performance [7].

The other factor that controls the complexity and the performance of the H.264/AVC codec is the number of block sizes used in the inter prediction process. Increasing the number of used block sizes results in better prediction and consequently higher compression performance at the expense of increased complexity. The complexity of motion estimation (ME) can be classified into different level of complexity, depending on the number of block size candidates used. In general, there are seven block sizes defined for inter-prediction in H.264/AVC.

In this paper, we analyze the effect of different coding parameters on the coding complexity using a set of training videos and propose a model for the relationship between coding configuration and coding complexity, and later this model is tested on a set of unseen test video set. The following subsections provide more details on our experiment settings and the proposed model.

A. Experiment Settings

In VSN applications, due to the limitations in energy and processing resources, less complex encoder configurations are used. To this end, we used baseline profile of H.264/AVC that is suitable for low complexity applications and uses only I and P frames (no B-frames) in our study. The other encoding parameters in our experiments include using context-adaptive variable-length coding (CAVLC) entropy coding and one reference frame, setting SR equal to 8, and disabling the rate distortion optimization (RDO), rate control, deblocking filter and Intra coding for P frames options. Furthermore, to have an objective measure for the encoding complexity, we use the instruction level profiler *iprof* [8], which provides us with the total number of instruction counts. The H.264/AVC reference software, JM version 18.2 is used in our experiments. Five representative videos from [9] are used in our study (BQMall, Traffic,

TABLE I
ME COMPLEXITY LEVEL (M_L)

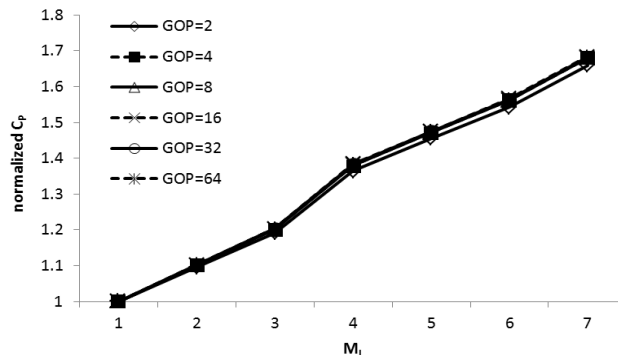| M_L	Block Size Candidates
1	SKIP, 16x16
2	SKIP, 16x16, 16x8
3	SKIP, 16x16, 16x8, 8x16
4	SKIP, 16x16, 16x8, 8x16, 8x8
5	SKIP, 16x16, 16x8, 8x16, 8x8, 8x4
6	SKIP, 16x16, 16x8, 8x16, 8x8, 8x4, 4x8
7	SKIP, 16x16, 16x8, 8x16, 8x8, 8x4, 4x8, 4x4

Fig. 2. Normalized C_P for different M_L for "BQMall" video

Race Horse, PeopleOnStreet and Vidyo1). To mimic a common VSN data, these sequences are downsampled to the common intermediate format (CIF) resolution (352x288 pixels) and also their frame rate was reduced to 15 frames per second (fps). The BQMall and Traffic video sequences are used as the training set for the model and the rest of videos as the test set.

B. Complexity Modelling

The coding process complexity of a video sequence (C_S) is formulated as follows:

$$C_S = C_I \cdot n_I + C_P \cdot n_P \qquad (1)$$

where C_I is the complexity to encode an I-frame, C_P is the complexity to encode a P-frame, n_I is the number of I-frames in the sequence and n_P is the number of P-frames in the sequence. For a video sequence with no scene change, the value of C_I can be considered almost constant. On the other hand, C_P depends on the complexity level of the ME process. In our study, the complexity level of ME process (called M_L) is classified based on the used block-size candidates in the encoding process as shown in Table I.

As illustrated in Fig. 2, the GOP size does not affect the normalized coding complexity of P frames at each M_L. Note that the complexity of coding P-frame (C_P) is normalized with respect to C_p when M_L is equal to one. Furthermore, as it can be seen from Fig. 3, the plot of normalized C_P for different training videos has the same slope but scaled by a constant. It can be seen from this figure that the normalized C_P for the Traffic video ranges from 1 to 1.485, which also

TABLE II
ME COMPLEXITY LEVEL (M_L) AND δ_{CP}

M_L	δ_{CP}
1	0
2	0.13
3	0.26
4	0.54
5	0.67
6	0.81
7	1
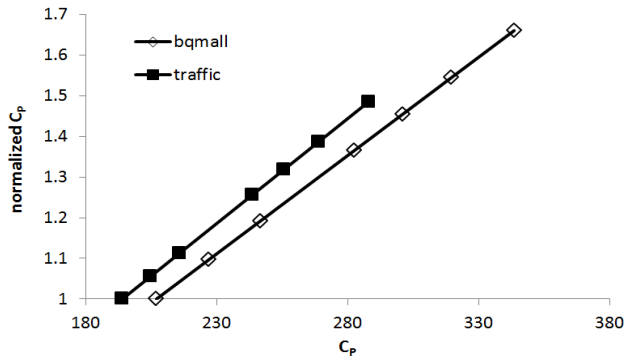
Fig. 3. Normalized C_P for $GOP=2$ of the training videos
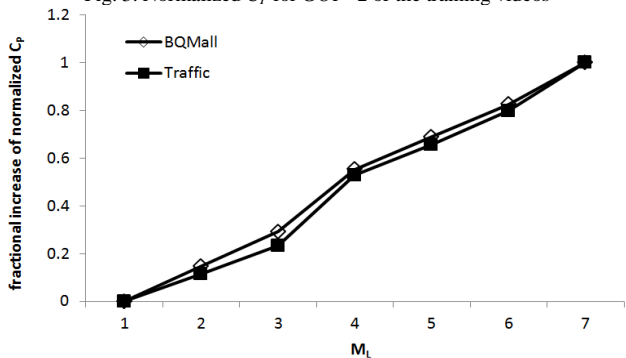
Fig. 4. Fractional increase of normalized C_P for the training videos

means that the normalized C_P range for this video is 0.485. On the other hand, the normalized C_P range for the BQMall video is equal to 0.66. Scaling the range of the normalized C_P to one, we can plot the fractional increase of normalized C_P as shown in Fig. 4. It is interesting to see that the increase of normalized C_P with respect to M_L is almost similar for both videos. We define δ_{CP} as the amount of increase normalized C_P at different M_L. δ_{CP} is calculated by averaging the values obtained in Fig. 4, as shown in Table II.

Another interesting observation is that, the value of range of normalized C_P shown in Fig. 2 is proportional to the value of $Cp_{M_L=1}$. Therefore, using the values obtained from the training videos, the range of normalized C_P values for a specific video sequence is calculated as:
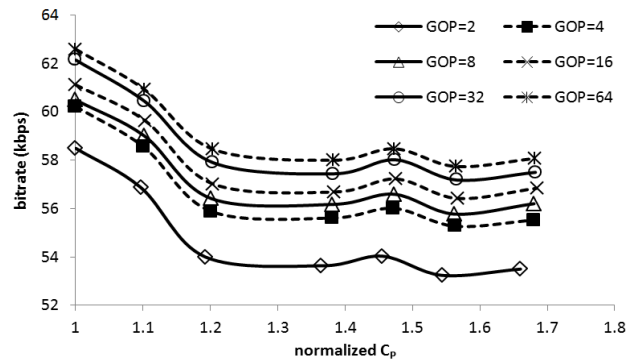
Fig. 5. Bitrate of a P-frame for different M_L of "BQMall" video

$$\omega_I = 0.0135 \cdot Cp_{M_L=1} - 2.13 \qquad (2)$$

Using ω_I, the complexity to encode a P-frame is formulated as:

$$Cp_{M_L=i} = Cp_{M_L=1} \cdot \left(1 + \delta_{CP}(i) \cdot \omega_1\right) \qquad (3)$$

Considering that $n_I = N/GOP$, where N is total number of frames and $n_P = N - N/GOP$, then the average complexity per frame is computed as follows:

$$C_f = (C_I + Cp_{M_L=1} \cdot (1 + \delta_{CP} \cdot \omega_1) \cdot (GOP - 1))/GOP \qquad (4)$$

C. Bitrate Modelling

The bitrate of the encoded video is modeled as $R = R_f \cdot F_r$, where R_f is the average bitrate of a frame and F_r is the frame rate. The total size of the encoded sequence (in bit) is then modeled as:

$$R_S = R_I \cdot n_I + R_P \cdot n_P \qquad (5)$$

where R_I is the average size of an I-frame and R_P is the average size of a P-frame. The value of R_P depends on the M_L and GOP used by the encoder.

Fig. 5 shows that, the value of R_P decreases as M_L increases. Therefore, for a certain GOP value, the R_P is modeled as:

$$R_{P_{GOP=i}} = \omega_{Ri} \cdot f(M_L) \qquad (6)$$

where ω_{Ri} is the bitrate of a P-frame when $GOP=i$ and $M_L=1$, and $f(M_L)$ is a decay function with respect to M_L, which is modeled using the generalized logistic function. The logistic function is a widely used sigmoid function for growth/decay modeling where the growth/decay is exponential at first, but eventually slower and then levels off. This matches the way R_P is reduced with the increase of
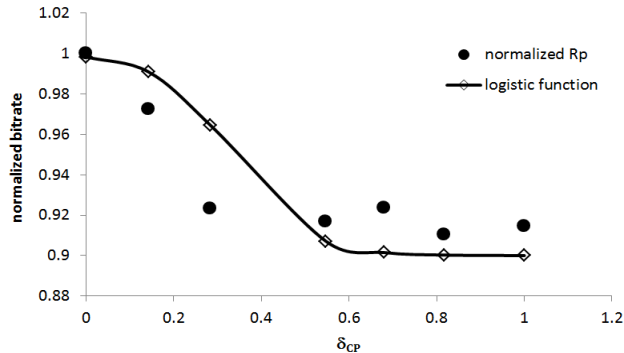
Fig. 6. The normalized bitrate ("BQMall", *GOP*=2) and the logistic function

M_L (see Fig. 5). The logistic function $f(M_L)$ used in our study is as follows:

$$f(M_L) = a + \frac{b-a}{1+e^{-c(x-d)}} \qquad (7)$$

where *a* and *b* indicate the minimum and maximum asymptote of the plot respectively, *c* is the growth rate, while *d* signify the time for maximum growth (see Fig. 6).

Furthermore, Fig. 5 also shows that the slope of the R_P plot for different *GOP* sizes is the same. Therefore, R_P is modeled equal to:

$$R_P = \omega_{Rp} \cdot f(M_L) + \omega_2 \cdot f(GOP) \qquad (8)$$

where ω_{Rp} is the bitrate of P-frame when *GOP*=2 and M_L=1, and ω_2 is the weight for *f(GOP)*. To obtain the parameters for the *f(M_L)*, we applied least mean square approach using the normalized R_P of training video sequences when *GOP*=2. Also to estimate *f(GOP)*, we applied curve fitting approach on the *Rp* values of training video sequences at different *GOP* size settings, and found that $\omega_2 \ln(GOP)$ provides a good estimate for *f(GOP)*. The value of ω_2 is estimated using least square regression from the training sequences. Assuming that the average bitrate of an I-frame is equal to R_I the average bitrate of a frame (R_f) is estimated as:

$$R_f = \frac{R_I}{GOP} + \omega_{RP} \cdot \left(0.92 + \frac{0.08}{(1+e^{3.56(6\cdot\delta_{CP}-0.84)})}\right) \cdot \frac{(GOP-1)}{GOP} \cdot$$
$$+ \omega_2 \cdot \ln(GOP) \cdot \frac{(GOP-1)}{GOP} \qquad (9)$$

D. Implementation of the Proposed Model

To implement the proposed model, we need to obtain several variables from each video sequence. To this end, we encode the first two frames of each video sequence. Assuming that there is no scene change in the video sequence, the bitrate of each I-frame will be almost similar. Therefore, R_I is assumed to be equal to the bitrate of the encoded first frame while ω_{Rp} is equal to the bitrate of the second frame. For the complexity modeling, the *iprof* tool will provide us with the complexity of encoding the first two frames of the video sequence, i.e., $C_{2\text{-frames}} = C_I + Cp_{M_L=1}$. Since we already have the value of R_I from encoding the first two frames of each test sequence, we can estimate the value of C_I of these sequences. The value of $Cp_{M_L=1}$ can then be calculated using $C_{2\text{-frames}} - C_I$. Consequently, the value of ω_I is calculated using (2).

To estimate the modeling error, the average percentage of complexity and bitrate error for GOP={1, 2, 4, 8, 16, 32, 64} and M_L={1, 2, 3, 4, 5, 6, 7} is calculated. As Table III shows, the average error for complexity modeling is less than or equal to 3.45% for the test video sequences, while the average error of bitrate modeling is less than or equal to 11.6% as reported in Table IV.

III. ENCODING AND TRANSMISSION POWER CONSUMPTION MODEL

The total power dissipation at a sensor node consists of the power consumption for encoding (P_e), transmission (P_t) and reception (P_r). P_e can be calculated as follows:

$$P_e = C_f \cdot F_r \cdot CPI \cdot E_c \qquad (10)$$

where *CPI* is the number of CPU cycles to perform one basic instruction and E_c is the energy depletion per cycle. The transmission power consumption is calculated as:

$$P_t = \sum (\alpha + \beta \cdot d^\eta) \cdot R \qquad (11)$$

where α is a constant coefficient related to coding and modulation, β is the amplifier energy coefficient, *d* is the transmission distance, η is path loss exponent and *R* is the bitrate. The reception power consumption is calculated as:

$$P_r = \sum \lambda \cdot R \qquad (12)$$

TABLE V. PARAMETERS USED.

Parameters	Description	value
α	Energy cost for transmitting 1 bit	0.5 J/Mb
β	Transmit amplifier coefficient	$1.3 \cdot 10\text{-}8$ J/Mb/m4
λ	Energy cost for receiving 1 bit	0.5 J/Mb
η	Path loss exponent	4
CPI	XScale average cycle per instruction [10]	1.78
E_c	Energy depleted per cycle for imote2 [6]	1.215 nJ

where λ is a constant coefficient representing energy cost for receiving 1 bit. Table V shows the parameters used for our experiments.

In this paper, we analyze a simple topology consisting of one video node and the sink. The total power consumption of a video node for different transmission distances for PeopleOnStreet video sequence is shown in Fig. 7. In this figure, we analyze two scenarios: a) the *GOP* size is fixed while the M_L varies, and b) the M_L is fixed while the *GOP* size changes. In Fig. 7a, the *GOP* size is set equal to eight and M_L changes. It is observed that for transmission distance less than 200m, the use of bigger M_L results in higher total power consumption. This result shows that varying M_L values do not significantly affect the trade-off between computation and communication. This trend is also seen in other test video sequences.

Fig. 7b shows the plot of total power consumption when M_L is equal to four and the *GOP* size changes. The figure shows that when the transmission distance is small, the configuration that leads to low power consumption is the one using smaller *GOP*. It means that the low encoding power consumption (due to the use of smaller *GOP*) is compensating the higher transmission power consumption (due to higher bitrate). However, when the transmission distance is large, the energy cost to transmit the data increased significantly. Therefore, we need to use the configuration with better compression performance, i.e., larger *GOP* size, to reduce the transmission energy consumption.

The trade-off between computation and communication can be clearly seen when the transmission distance is less than 100m as shown in Fig. 8. However, it can be seen that the transmission distance at which the use of bigger *GOP* minimizes power consumption is content dependent. For example, in the case of PeopleOnStreet video sequence, using *GOP* equal to one will minimize the total power consumption when the transmission distance is less than 63m (see Fig. 8a). However, for the RaceHorses video sequence, the use of *GOP* equal to one will minimize total power consumption when the transmission distance is less than 88m (see Fig. 8b).
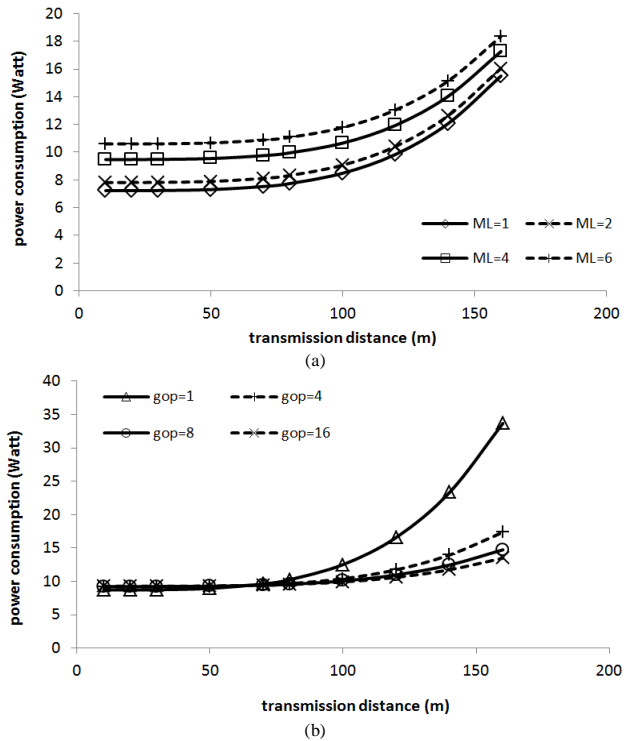

Fig. 7. Total power consumption for different transmission distance: (a) *GOP*=8 and varying M_L (b) M_L=4 and varying *GOP* sizes
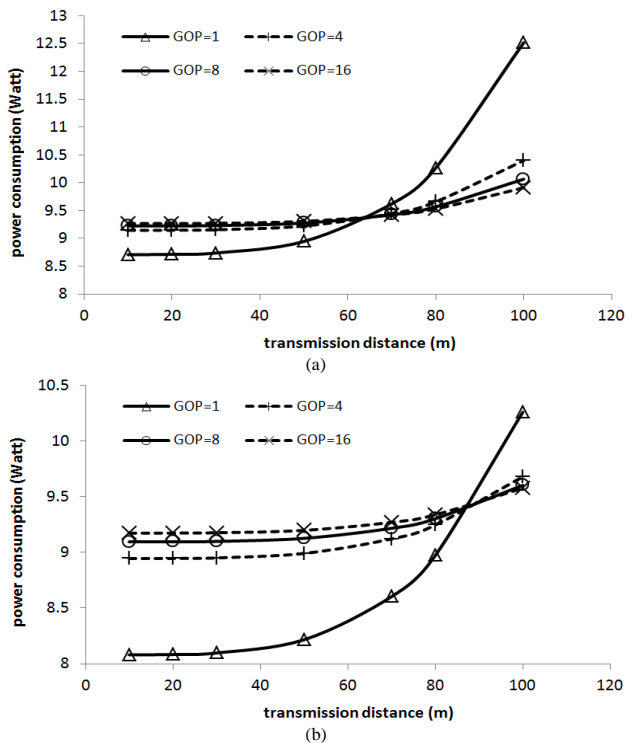

Fig. 8. Total power consumption for transmission distance less than 100m (M_L=4 and varying *GOP* sizes): (a) PeopleOnStreet sequence(b) RaceHorses sequence

IV. CONCLUSION

In this paper, we propose the encoding complexity and bitrate model of H.264-based video sensor networks. The experimental results show that the proposed complexity model provides a very small prediction error (less than or equal to 3.45%), while the bitrate modeling error is from 8.57% to 11.6% for the video sequences tested. The proposed model is used to show the trade-off between encoding and communication that can be exploited to minimize the total power consumption of VSNs.

REFERENCES

[1] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury, "A survey on wireless multimedia sensor networks," Computer Networks: The International Journal of Computer and Telecommunications Networking, vol. 51, no. 4, pp. 921–960, Mar. 2007.

[2] T. D. R¨aty, "Survey on Contemporary Remote Surveillance Systems for Public Safety," Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, vol. 40, no. 5, pp. 493–515, Sep. 2010.

[3] I. E. Richardson, The H.264 Advanced Video Compression Standard, Second Edition. John Wiley & Sons, Ltd, 2010.

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

[5] H. K. Zrida, A. C. Ammari, M. Abid, and A. Jemai, "Complexity/Performance Analysis of a H.264/AVC Video Encoder," in Recent Advances on Video Coding, InTech.

[6] J. J. Ahmad, H. A. Khan, and S. A. Khayam, "Energy efficient video compression for wireless sensor networks," presented at the Information Sciences and Systems, 2009. CISS 2009. 43rd Annual Conference on, Baltimore, MD, 2009, pp. 629 – 634.

[7] B. A. B. Sarif, M. T. Pourazad, P. Nasiopoulos, and V. C. M. Leung, "Encoding and communication energy consumption trade-off in H.264/AVC based video sensor network," presented at the Accepted for publication IEEE World of Wireless, Mobile and Multimedia Networks, WoWMoM'13, 2013, pp. 1–6.

[8] P. M. Kuhn, "A Complexity Analysis Tool: iprof (version 0.41)," ISO/IEC JTC1/SC29/WG11/M3551, Jul-1998.

[9] ISO/IEC JTC1/SC29/WG11, "Joint Call for Proposals on Video Com-pression Technology." Jan-2010.

[10] D. Chinnery and K. Keutzer, Closing the Power Gap between ASIC & Custom: Tools and Techniques for Low Power Design, 1st edition. Springer, 2007.