Modeling and Adapting JPEG to the energy requirements of VSN

Abdelhamid Mammeri*, IEEE student member, Ahmed Khoumsi†, Djemel Ziou† and Brahim Hadjou*  
*Department of Electrical and Computer Engineering  
†Department of Computer Science  
University of Sherbrooke, Quebec, Canada, J1K 2R1  
Email: Firstname.Lastname@USherbrooke.ca

Abstract—We address the problem of modeling and adapting JPEG to the energy requirements of Visual Sensor Networks (VSN). For JPEG modeling purposes, we develop a simplified high-level energy consumption model for each stage of JPEG-like scheme, which can be used to roughly evaluate the energy dissipated by a given visual sensor. This model is based on the basic operations needed at each stage of JPEG, and it does not take into account the complexity of implementation. For JPEG adaptation, we propose to process only a reduced part of each block of $8 \times 8$ DCT coefficients of the target image, which minimizes the dissipated energy and maximizes the system lifetime, while preserving an adequate image quality at the sink.

I. INTRODUCTION

Visual Sensor Networks (VSN) are a special type of Wireless Sensor Networks (WSN) that contain Visual Sensors (VS) incorporating image retrieving, processing and communication capabilities. VSN are involved in many domains such as video-surveillance and object detection/tracking. One of the biggest challenges that face VSN is the huge amount of information processed and sent by each VS, which requires important resources. Typically, visual information needs to be compressed using some standards such as JPEG or JPEG2000 to save energy. Unfortunately, these standards are energy consuming in data processing and generate compressed information that still contains redundant data which might be compressed further.

Motivated by the node resource constraints in VSN, in this work, we address the problem of modeling and adapting JPEG to the energy requirements of such networks. First, to evaluate the consumed energy by one VS, we develop an energy model for JPEG-like scheme. To the best of our knowledge, no such energy model has been previously proposed. This model is quite high-level and is based on the basic operations needed at each stage of JPEG, and it does not take into account the complexity of implementation. Our model can serve as a mean to simulate and to roughly evaluate the energy dissipated by one VS when executing JPEG or its adapted version. Finally, the adaptation of JPEG is made with respect to the following metrics: the image quality, the consumed energy and the cluster lifetime. The adapted version of JPEG, which we call Triangular JPEG (T-JPEG), is performed by exploring the characteristics of the energy compaction property of the Discrete Cosine Transform (DCT). Indeed with JPEG, the bulk energy after DCT transformation is concentrated in the first few low frequencies coefficients of DCT blocks. We will select only these coefficients from each DCT block, in order to minimize the number of operations at each stage of the compression scheme. The aim of T-JPEG is the minimization of the energy consumed at each VS while keeping a relatively good image quality, which is acceptable for some VSN applications that face severe energy constraints. For transmission purposes, a simplified routing module is considered.

The concept of reducing the number of $8 \times 8$ DCT coefficients within each block of the decomposed image was previously used in [1] and [2]. The authors in [1] investigated the minimization of such operations by choosing a squared reduced portion of the block of $8 \times 8$ DCT coefficients. Their idea was used in centralized wireless multimedia networks which differ from VSN requirements. This concept was also used in [2] to design an energy-aware VLSI system for portable devices to compress an image. As in [1], the authors of [2] do not use an adequate selection of AC coefficients, nor present any analytical analysis of the energy gain.

The rest of this paper is organized as follows. The network model and the radio transceiver energy model are presented in Section II. In Section III, we present the energy model of JPEG. We introduce in Section IV, T-JPEG and its energy model. In Section V, we present the routing module. In Section VI, we validate our results by a set of simulations. We summarize and present future directions in Section VII.

II. SYSTEM MODEL

We consider a two-tiered VSN divided into several clusters, where each cluster contains several VS nodes and a single cluster head (CH) distributed around strategic locations. Each node in the network is assumed to consist of a low-power imaging sensor connected to an embedded sensor (e.g., Mica2). Within a cluster, each VS is responsible for capturing, encoding and transmitting relevant images from areas of interest to the CH of the cluster. The following assumptions are used in our network model. All VS nodes are suitably distributed in a strategic area to ensure that the network is fully connected. Each node communicates with its neighbours within its transmission range $d$. All nodes are immobile and energy constrained. Moreover, each VS node knows its location through a localization algorithm as in [3].
or some related GPS techniques, and keeps information about its neighbors such as ID number and the remaining energy.

The power consumption model for the radio is similar to the one proposed in [4]. The energy dissipated by a given VS for sending a single bit is $e_{Tx} = e_c + a d^\alpha$, and the consumed energy in reception per bit is $e_{Rx} = e_c$. Where $e_c$ is the energy consumed by the transmit amplifier per bit over a distance of 1 meter, $e_c$ is the energy dissipated by the transmitter electronics per bit, $\alpha \in [2, 4]$ is the path loss exponent and $d$ is the distance between the sender and the receiver [5]. Therefore, the total energy consumed for transmitting a bit between two nodes is $e_b = e_{Tx} + e_{Rx}$. In our work, we consider the linear multihop scenario as in [4]. For $n$ hops between the source and the sink, the multihop energy per bit can be stated as:

$$e_b = ne_{Tx} + (n-1)e_{Rx} \quad (1)$$

III. ENERGY CONSUMPTION MODEL OF JPEG

We develop a high-level power consumption model for JPEG [6] stages (Figure 1), which can be used to roughly evaluate the energy $E_p$ dissipated by a given VS when executing JPEG. The following assumptions are used to estimate this energy. The suggested model is quite high-level, and is only based on the basic operations needed at each stage of the compression scheme, and does not take into account the complexity of implementation (where several optimization techniques can be found in the literature). Moreover, some parameters like memory access are not considered.

![JPEG Compression Scheme](image)

**Fig. 1.** Lossy JPEG compression scheme

$E_p$ can be formulated as follows, where $E_{dct}$, $E_q$, $E_z$, $E_{rle}$ and $E_{huf}$ represent the energies consumed at 2D DCT, quantizing, zigzagging, RLE and Huffman (entropy) encoding stages, respectively.

$$E_p = E_{dct} + E_q + E_z + E_{rle} + E_{huf} \quad (2)$$

A. 2D DCT Energy model

We develop an energy consumption model for 2D DCT transformation based on the 2D DCT equation given by: [6]

$$F(u, v) = \frac{C(u)C(v)}{\sqrt{2k}} \times \sum_{x=0}^{k-1} \sum_{y=0}^{k-1} f(x, y) \cos \frac{(2x + 1)u\pi}{2k} \cos \frac{(2y + 1)v\pi}{2k} \quad (3)$$

Where $u, v$ are the discrete frequency variables such that $0 \leq u, v \leq k-1$, $k$ is the block size; $f(x, y)$ is the grey level of one pixel at $(x, y)$ in the $N \times N$ image, where $0 \leq x, y \leq k-1$; $F(u, v)$ is the coefficient of the point $(u, v)$. The coefficients $C(u)$ and $C(v)$ are equal to $\frac{1}{\sqrt{2}}$ if $(u, v) = 0$, and to $1$ if $1 \leq u, v \leq k-1$. Based on Eq. 3, the 2D DCT can be modeled as two $k \times k$ matrix multiplications:

$$F_{(k \times k)} = A_{(k \times k)} F_{(k \times k)} A^T_{(k \times k)} \quad (4)$$

Where the index $(k \times k)$ indicates a size of matrix. $F$ is the $k \times k$ matrix whose coefficients are $F(u, v)$, $P$ is the matrix of pixels of the original image and $A^T$ is the transpose of $A$ whose coefficients are: $A(u, v) = \frac{1}{\sqrt{k}}$ if $u = 0$ and $A(u, v) = \sqrt{\frac{2}{k}} \cos \left(\frac{(2u+1)\pi}{2k}\right)$ if $1 < u < k-1$.

Each matrix product $k \times k$ of Eq. 4 consists in computing $k^2$ coefficients. Each coefficient necessitates $k$ multiplications and $(k-1)$ additions. Therefore, the energy dissipated for the two matrix products is $2k^2(k e_{mult} + (k-1)e_{add})$, where $e_{mult}$ and $e_{add}$ represent the energy consumption for mult and add instructions, respectively. For an image of size $N \times N$ having $(\frac{N}{k})^2$ blocks, the total energy dissipated in this stage is:

$$E_{dct} = \left(\frac{N}{k}\right)^2 e_{dct} = \left(\frac{N}{k}\right)^2 2k^2(k e_{mult} + (k-1)e_{add}) \quad (5)$$

B. Quantization energy model

The energy consumption by the quantization stage can be expressed as a function of $F(u, v)$ and $Q(u, v)$. Where $F(u, v)$ is defined by Eq. 3 and $Q(u, v)$ is a coefficient of the luminance quantization matrix $Q$ defined as in [6] for $k = 8$. For each block, quantization consists in computing the coefficients $F^Q(u, v) = \text{IntegerRound}(F(u, v)Q(u, v))$, for $0 \leq u, v \leq k-1$. Therefore, there are $k^2$ divisions and $k^2$ round operations per block and the dissipated energy is thus $e_q = k^2(e_{div} + e_r)$, where $e_{div}$ and $e_r$ are the energy consumption for div and round instructions, respectively. For one image of size $N \times N$ having $(\frac{N}{k})^2$ blocks, the total energy consumption related to the quantization stage is:

$$E_q = \left(\frac{N}{k}\right)^2 e_q = \left(\frac{N}{k}\right)^2 k^2(e_{div} + e_r) \quad (6)$$

C. Zigzagging energy model

This stage can be interpreted as a simple rearrangement of the $(k^2-1)$ AC coefficients in zigzagging way, from low to high frequencies. Each AC coefficient is then shifted to its new position. The zigzagging energy consumption on each block can be calculated as: $e_z = (k^2-1)e_{sh}$, where $e_{sh}$ represents the energy consumption of shift process. For one input image of size $N \times N$, the total energy consumption related to the zigzagging stage is:

$$E_z = \left(\frac{N}{k}\right)^2 e_z = \left(\frac{N}{k}\right)^2 (k^2-1)e_{sh} \quad (7)$$

D. RLE energy model

Run length encoding of the $(k^2-1)$ AC coefficients can be interpreted mainly as a rearrangement of the zigzagged sequence as follows, where $0seq$ denotes a sequence of zeros.

1) If the zigzagged sequence terminates by a $0seq$, the latter is replaced by $(0, 0)$, meaning End Of Block (EOB). For the other $0seq$s, we proceed as explained in the following points 2-4.
2) Each $0\text{seq}$ of length $L$ is decomposed into $(L \div 16)$ $0\text{seqs}$ of length 16 and a last $0\text{seq}$ of length $(L \mod 16)$. Note that after this decomposition, each $0\text{seq}$ of length < 16 is followed by a non-zero coefficient.

3) Each $0\text{seq}$ of length 16 is replaced by (15,0), meaning that a zero is preceded by 15 zeros.

4) Each $0\text{seq}$ of length $r < 16$ and the non-zero value $X$ that follows are replaced by $(r,X)$, meaning that $X$ is preceded by $r$ zeros.

To compute the energy dissipated by one block, we consider the following parameters. $e_z$ is the consumed energy for checking whether an AC coefficient is null. $e_{wr}$ is the dissipated energy for writing one $(r,X)$ value (see above Points 3-4). $e_{inc}$ is the dissipated energy for incrementing the counter of the number of 0 in each $0\text{seq}$. $e_{res}$ is the consumed energy for resetting the counter. $N_{0\text{seq}}$ is the number of $0\text{seqs}$ inside one $k \times k$ block. Let us denote these $0\text{seqs}$ by $0\text{seq}[1], 0\text{seq}[2], \ldots, 0\text{seq}[N_{0\text{seq}}]$, and let $length(i)$ be the number of zeros in $0\text{seq}[i]$. Therefore, the total number of zeros inside one block is $N_0 = \sum_{i=1}^{N_{0\text{seq}}} length(i)$.

The total energy can be estimated as follows. The values of the $(k^2-1)$ AC coefficients are checked, dissipating the energy $(k^2-1)e_z$. The value $((k^2-1)-(N_0+N_{0\text{seq}}))$ is the number of non-zero AC coefficients which are not preceded by a $0\text{seq}$, and each coefficient $X$ of this category necessitates to write $(0,X)$, dissipating in total the energy $((k^2-1)-(N_0+N_{0\text{seq}}))e_{wr}$. For each non-zero value and the $0\text{seq}[i]$ that precedes it, we consider $p_i = length(i) \text{div} 16$ and $q_i = length(i) \text{mod} 16$. After decomposition (see above Point 2), we obtain $(p_i+1)0\text{seq}$. Each of these $0\text{seq}$ necessitates to reset the counter and to write $(r,X)$, dissipating in total the energy $(p_i+1)(e_{wr}+e_{ws})$. Each of the first $p_i \times 0\text{seq}$ (whose length is 16) necessitates to increment the counter 15 times, and the last $0\text{seq}$ (whose length is $q_i$) necessitates to increment the counter $q_i-1$ times, dissipating in total the energy $(15p_i+q_i-1)e_{inc}$. Therefore, the energy $e_{0\text{seq}}(i)$ for processing the decomposed $0\text{seq}[i]$ can be formulated as follows:

$$e_{0\text{seq}}(i) = \begin{cases} (p_i+1)(e_{wr}+e_{ws}) + (15p_i+q_i-1)e_{inc}, \\ e_{wr}, \quad \text{if } (i < N_{0\text{seq}}) \lor (\text{Last_component} \neq 0) \\ e_{ws}, \quad \text{else} \end{cases}$$

The total energy dissipated inside one block of $k \times k$ is:

$$e_{rlc} = (k^2-1)e_z + ((k^2-1)-(N_0+N_{0\text{seq}}))e_{wr} + \sum_{i=1}^{N_{0\text{seq}}} e_{0\text{seq}}(i)$$

For one image having $N \times N$ pixels, the energy dissipated by RLE stage is:

$$E_{rlc} = \sum_{j=1}^{N^2} e_{rlc}$$

Where $e_{rlc}$ is the energy consumed by the $j^{th}$ block computed using Eq. 9.

E. Huffman encoding energy model

For Huffman encoding, we use the same Huffman tables as in [6]. With JPEG, AC and DC coefficients are encoded separately as follows.

a) AC coefficients: Huffman encoding of AC coefficients can be summarized in two steps. In the first step, the $m_j$ pairs $(r_i,X_i)$ computed in RLE stage for the $j^{th}$ block are put into categories, where $i = 1 \ldots m_j$ and $j = 1 \ldots (\frac{N}{16})^2$ (number of blocks). In JPEG, for each $(r_i,X_i)$ we encode only the right value $X_i$, except the pairs that are special markers like (0,0) or (15,0). Let $\text{Cat}(X_i)$ be the category value of $X_i$, and $\text{Bin}(X_i)$ is its bit-coded representation (both extracted from the category table [6]). In JPEG, $X_i$ is encoded as $(\text{Cat}(X_i), \text{Bin}(X_i))$. Hence, the found string pairs for the $j^{th}$ block are: $(r_1,\text{Cat}(X_1)), \text{Bin}(X_1); (r_2,\text{Cat}(X_2)), \text{Bin}(X_2); \ldots (r_{m_j},\text{Cat}(X_{m_j})), \text{Bin}(X_{m_j})$. The second step consists in Huffman encoding the byte $(r_i,\text{Cat}(X_i))$ by looking in AC Huffman table for its correspondent code, which we note $\text{Huff}_i$. After these two steps and for the $j^{th}$ block having $m_j$ pairs, we obtain the following stream of AC coefficients which is written in the JIF file: $[\text{Huff}_1 \text{Bin}(X_1) \text{Huff}_2 \text{Bin}(X_2) \ldots \text{Huff}_{m_j} \text{Bin}(X_{m_j})]$.  

b) DC coefficients: In JPEG, DC coefficients are Huffman encoded differently than AC coefficients. Huffman encoding of DC coefficients can be summarized in two steps. In the first step, the computed difference $\text{Diff} = D_{C_i} - D_{C_{i-1}}$ is put into a category, where $j = 1 \ldots (\frac{N}{16})^2$, and $D_{C_0} = 0$. Let $\text{Cat}(\text{Diff})$ be the category value of $\text{Diff}$, and $\text{Bin}(\text{Diff})$ is its bit-coded representation, both extracted from the category table [6]. In JPEG, $\text{Diff}$ is encoded as $(\text{Cat}(\text{Diff}), \text{Bin}(\text{Diff}))$. The second step consists in Huffman encoding the value $\text{Cat}(\text{Diff})$ by looking in DC Huffman table for its correspondent code, which we note $\text{Huff}$. From these steps, $\text{Diff}$ will be coded as $[\text{Huff}(\text{Cat}(\text{Diff})) \text{Bin}(\text{Diff})]$. Therefore, the final bit stream of DC and AC coefficients written in the JPEG file for the $j^{th}$ block is given by:

$$[\text{Huff}(\text{Cat}(\text{Diff})) \text{Bin}(\text{Diff})] \text{Huff}_1 \text{Bin}(X_1) \text{Huff}_2 \text{Bin}(X_2) \ldots \text{Huff}_{m_j} \text{Bin}(X_{m_j})]$$

Following the two steps cited previously for Huffman encoding of AC coefficients, the energy dissipated for the $j^{th}$ block is $e_{\text{huf}}^{ac} = m_j(e_{\text{fetch}} + e_{\text{fetch}} + e_{\text{ws}})$, where $m_j$ is the number of pairs $(r_i,X_i)$ within the $j^{th}$ block, except the pairs that are special markers like (0,0) or (15,0); $e_{\text{ws}}$ is the energy required to write a stream of bits in the JPEG file; the first $e_{\text{fetch}}$ in the equation $e_{\text{huf}}^{ac}$ is dissipated when we look in the category table for the representation of $X_i$, whereas the second $e_{\text{fetch}}$ is consumed when we look in the final step of Huffman encoding of the byte $(r_i,X_i)$. The energy dissipated per image (for AC coefficients) is $E_{\text{huf}}^{ac} = \sum_{j=1}^{(\frac{N}{16})^2} e_{\text{huf}}^{ac}$. While for DC coefficients, the energy dissipated by the $j^{th}$ block is $e_{\text{huf}}^{dc} = (e_{\text{fetch}} + e_{\text{fetch}} + e_d + e_{\text{ws}})$. Where $e_d$ is the energy required to compute the difference $\text{Diff}_j$; the first $e_{\text{fetch}}$ in equation $e_{\text{huf}}^{dc}$ is consumed when we look in the category table for the representation of $\text{Diff}_j$, whereas the second $e_{\text{fetch}}$
is dissipated when we look in DC Huffman table for the representation of \( \text{Huff(cat(Diff_j))} \). The energy dissipated per image for DC coefficients is \( E_{hufDC} = \sum_{j=1}^{(\frac{N}{2})^2} e_{hufDC(j)} \).

Finally, the total energy dissipated per image for both DC and AC coefficients is the sum of \( E_{hufDC} \) and \( E_{hufAC} \):

\[
E_{huf} = \sum_{j=1}^{(\frac{N}{2})^2} (e_{hufDC(j)} + e_{hufAC(j)})
\] (11)

IV. T-JPEG AND ITS ENERGY MODEL

Triangular JPEG (T-JPEG) means the adaptation version of JPEG incorporating the Reduced Block Size notion, which consists in processing only the upper-left triangular \( E \) we have done with JPEG, and quantify the gain resulting from energy consumed for each stage of the compression scheme as \( E \) therefore clear that the energy dissipated energy, while keeping an acceptable image quality increases, both the quality of the received image and the energy consumed by one node increase. In the following paragraphs, given \( \rho \), we present the energy model of T-JPEG.

Fig. 2. Triangular selection of AC coefficients (\( \rho = 4 \)).

We have seen that with JPEG, for each \( k \times k \) DCT block, \( k^2 \) coefficients are processed and transmitted. The gain in energy using T-JPEG, is basically due to the fact that \( C_\rho = \rho \rho \rho (k^2, assuming k > 1 ) \) are processed and transmitted. It is therefore clear that the energy \( E_\rho \) dissipated by T-JPEG is lower than \( E_\rho \) dissipated by JPEG. Let us now evaluate the energy consumed for each stage of the compression scheme as we have done with JPEG, and quantize the gain resulting from the passage from JPEG to T-JPEG. \( E_\rho \) is defined as follows:

\[
E_\rho = E_{dct_\rho} + E_{q_\rho} + E_{z_\rho} + E_{rle_\rho} + E_{huf_\rho}
\] (12)

Where \( E_{dct_\rho}, E_{q_\rho}, E_{z_\rho}, E_{rle_\rho}, \) and \( E_{huf_\rho} \) are the energies consumed at 2D DCT, quantizing, zigzagging, RLE and Huffman stages, respectively. Let us develop the energy model of T-JPEG, being inspired from the model of all stages of JPEG. In DCT stage, if we consider the matrix product of Eq. 4, we are then interested only by the upper-left triangular portion of size \( \rho \) of \( F_{(k \times k)} \), that is, by the coefficients \( F(u, v) \) such that \( u + v \leq \rho + 1 \). For this reason, it is sufficient to consider only the matrix product of Eq. 13, where the expression \( U_{(a \times b)} \) means the upper-left rectangular (a lines; b rows) portion of a matrix \( U_{(k \times k)} \):

\[
F_{(\rho \times \rho)} = A_{(\rho \times k)} P_{(k \times \rho)} A^T_{(\rho \times \rho)}
\] (13)

The first product \( A_{(\rho \times k)} P_{(k \times \rho)} \) consists in computing \( kp \) coefficients, each coefficient necessitating \( k \) multiplications and \( (k-1) \) additions. Let \( R_{(\rho \times \rho)} \) be the result of this first product. For the second product \( R_{(\rho \times \rho)} A^T_{(\rho \times \rho)} \), we need to compute only \( C_\rho \) coefficients, each of them necessitating \( k \) multiplications and \( (k-1) \) additions. Therefore, the total dissipated energy is:

\[
(kp + C_\rho) (ke_{mul} + (k - 1) e_{add})
\] (14)

For Eqs. 5 and 14, the gain in energy for DCT stage is

\[
\frac{E_{dcgan}}{E_{dcgan}} = \frac{kp + C_\rho}{2k^2} = 0.25 p + 0.5 \phi \frac{1}{k} \leq 1, \quad \text{where } \phi = \frac{1}{2k}
\]

For the following three stages (quantization, zigzagging, RLE), we simply replace \( k^2 \) by \( C_\rho \), in Eqs. 6-10, since we consider \( C_\rho \) instead of all \( k^2 \) coefficients of a DCT block. We obtain Eqs. 15-19.

In quantization stage, the energy \( E_q \) (Eq. 6) obtained with JPEG becomes \( E_{q_\rho} \) (Eq. 15) with T-JPEG:

\[
E_{q_\rho} = (\frac{N}{k})^2 C_\rho (e_{div} + e_r)
\] (15)

From Eqs. 6 and 15, the gain in energy for the quantization stage is

\[
\frac{E_{q_\rho}}{E_{qgan}} = C_\rho = \rho \rho \rho \rho \rho < 1
\]

In zigzagging stage, the energy \( E_z \) (Eq. 7) obtained with JPEG becomes \( E_{z_\rho} \) (Eq. 16) with T-JPEG:

\[
E_{z_\rho} = (\frac{N}{k})^2 (C_\rho - 1) e_{sh}
\] (16)

The gain in energy for zigzagging stage is computed from Eqs. 7 and 16 and is

\[
\frac{E_{z_\rho}}{E_{zgan}} = C_\rho = \frac{\rho \rho \rho - 1}{\rho \rho \rho} = \frac{(\rho - 1)(\rho + 2)}{\rho} \leq 1
\]

In RLE stage, the energy \( e_{rle} \) (Eq. 9) obtained with JPEG becomes \( e_{rle_\rho} \) (Eq. 17) with T-JPEG:

\[
e_{rle_\rho} = (C_\rho - 1) e_z + (C_\rho - 1) \sum_{i=1}^{N_{seq}} \sum_{i=1}^{N_{seq}} \sum_{i=1}^{N_{seq}} e_{0seq}(i)
\] (17)

Where,

\[
e_{0seq}(i) = \begin{cases} (p_i + 1)(e_{ur} + e_{rinc}) + (15p_i + q_i - 1)e_{inc}, & \text{if } i < N_{seq} \land (\text{Last_component } \neq 0) \\ e_{ur}, & \text{else} \end{cases}
\] (18)

Since \( C_\rho < k^2, N_{seq} < N_0, N_0 < N_0 \) and \( N_{seq} < N_{seq} \) we obtain

\[
e_{rle_\rho} < e_{rle}, \text{ where } N_{00} \text{ is the total number of zeros, and } N_{0seq} \text{ is the number of } 0\text{seqs, inside one triangle having } C_\rho \text{ coefficients. If we consider one image of } N \times N \text{ pixels, the energy } E_{rle} \text{ (Eq. 10) obtained with JPEG becomes } E_{rle_\rho} \text{ (Eq. 19) with T-JPEG, where } e_{rle_\rho} \text{ is the energy consumed by the } j^{th} \text{ block computed using Eq. 17.}

\[
e_{rle_\rho} = \sum_{j=1}^{(\frac{N}{2})^2} e_{rle_\rho(j)} < E_{rle}
\] (19)
For Huffman stage, it is obvious to note that the number of pairs \((r_i, X_i)\) resulting from RLE stage with T-JPEG are smaller or equal to those obtained with JPEG. Hence, \(E_{huf}^p\) dissipated with T-JPEG during Huffman stage is smaller or equal to \(E_{huf}\) (Eq. 11) obtained with JPEG, i.e.,

\[
E_{huf}^p \leq E_{huf}
\]  

(20)

From Eqs. 14-20, we deduce that the total energy \(E_p^f\) dissipated with T-JPEG is therefore given by the following equation, and is strictly smaller than the energy \(E_p\) consumed by JPEG, i.e.,

\[
E_p^f = E_{dct}^p + E_q^p + E_z + E_{rle}^p + E_{huf}^p < E_p.
\]

V. ROUTING MODULE

We used a simple routing scenario illustrated in Figure 3 and summarized as follows. First, after receiving the image query \(I_q\), the source node \(S\) takes an image \(I\), and compresses it using JPEG or T-JPEG. After that, \(S\) sends its packets to its cluster head \(CH_1\). Then, based on the energy of neighboring nodes, \(CH_1\) selects from its neighboring table, the set of nodes \(S_{nf}\) (in its cluster) that are able to forward the received packets. If \(S_{nf}\) is empty, i.e., there is no eligible node, \(CH_1\) forwards the received packets directly to the CH of the next cluster, thus consuming more energy. After that, each node \(n_f\) from \(S_{nf}\) sends the received packets to a node \(n_{next}\) belonging to the neighbor cluster. In its turn, \(n_{next}\) forwards the received packets to its cluster head \(CH_2\). If there is no node of the neighbor cluster inside the range of transmission of \(n_f\), the packets will be sent directly to \(CH_2\) (cluster head of the neighboring cluster), thus consuming more energy. This process will continue, until the reception of all packets by the base station.

![Fig. 3. Routing module](image)

VI. SIMULATIONS

Several simulations are conducted using Matlab. We study the effects of varying \(\rho\) on both the energy consumed by a VS node and the quality of the received image at the sink. After that, we compare JPEG with T-JPEG in VSN according to the following performance metrics:

- The communication and compression energy.
- The image quality measured by the peak signal-to-noise ratio (PSNR). For image quality requirements for VSN, we use the same quality levels suggested in [6], which may serve only as a guideline, since the quality can vary significantly according to image characteristics. These quality levels are: 0.25-0.50 bit/pixel: moderate to good quality; 0.50-0.75 bit/pixel: good to very good quality; 0.75-1.50 bit/pixel: excellent quality; Over 1.50 bit/pixel: usually indistinguishable from original.

- The notion Cluster lifetime is introduced instead of the commonly used network lifetime. The cluster is alive if there is at least one active sensor inside the cluster. The cluster lifetime is closely related to \(\rho\). If \(\rho\) is high, the source node consumes more energy, which reduces the cluster lifetime.

We consider the network model as described in Section II. The sensor network, formed by 300 nodes, is placed in a square region of size \(50 \times 50\ m^2\). We suppose that the number of sensors per cluster varies between 6 and 20 nodes, and each node has an initial energy \(E_0 = 10\ joules\). A node is considered non-functional if its energy level \(E_l\) reaches 0, and cannot take an image if \(E_l < E_p\) (Eq. 2). The node communication range \(d\) is fixed at 10 m. The values of \(e_a\), \(e_e\) and \(\alpha\) of the radio transceiver model are \(e_a = 100.10^{-12}\ Joule/bit/m^2\), \(e_e = 50.10^{-9}\ Joule/bit\) and \(\alpha = 2.5\) [7]. The size of data packets is 250 bytes.

To evaluate the processing energy consumed by JPEG and T-JPEG, we adopt the parameters of Mica2. From Equations 1, 2 and 12, technical documentation [8] and some experiences [9], we compute the total energy dissipated by JPEG and T-JPEG. For space limitation, we do not present the numerical computations. The reader is invited to consult the references cited in this section.

A. Effect of varying \(\rho\) on VS energy and image quality

In this simulation, several images are tested to show the two effects (Energy-effect/Quality-effect) of varying \(\rho\) on the source node energy and the image quality. The energy-effect means that the energy consumed by one source node for processing an image increases with \(\rho\). This is explained by the fact that the number of DCT coefficients processed increases with \(\rho\), which necessitates more processing energy. The quality-effect means that the received image quality at the sink raises with \(\rho\). For image quality levels, we use those presented in Section VI, where the user at the sink selects which value of \(\rho\) can be used. For example, for image Lena and for moderate-good image quality (0.25-0.50 bit/pixel), the value of \(\rho\) can be set to 2, 3 or 4, and hence we process only \(\frac{2(2+1)}{2} = 3\), \(\frac{3(3+1)}{2} = 6\), or \(\frac{4(4+1)}{2} = 10\) coefficients, respectively, instead of processing the whole block. And we dissipate only 1.19mJ (0.26 bpp, PSNR = 27.33), 2.21mJ (0.39 bpp, PSNR = 29.85) or 3.02mJ (0.48 bpp, PSNR = 31.92) for 2, 3 or 4 coefficients respectively, instead of dissipating 11.98mJ for the whole block.

B. Effect of varying \(\rho\) on the performance metrics

In this section, we simulate and study the effect of varying \(\rho\) on: 1) the energy dissipated through the path between the source and the sink computed using Eqs. 1 and 2 for JPEG, and Eq. 1 and 12 for T-JPEG, and 2) the cluster lifetime. That is, in T-JPEG we replace \(\rho\) by different values between 1 and 8, and analyze the effect of this replacement on these
performance metrics. We use the notation T-JPEG(\(\rho\)) to mean that T-JPEG is used for a given value \(\rho\), for example T-JPEG(5) means that T-JPEG is used for \(\rho = 5\). We suppose that the distance between the source and the sink takes several hops, from 1 to 30. In our simulations, we focus on unicast traffic pattern (Figure 3). We consider the same example of Lena image, and we compute the energy dissipated through the path used during a whole cycle\(^1\) to send this image using JPEG and T-JPEG for each value of \(\rho\) = 1 to 8. We note that for \(\rho = 8\), the number of processed coefficients is 36 and not 64 coefficients, i.e. T-JPEG(8) \(\neq\) JPEG. We consider different numbers of hops between the source and the destination. For a given number of hops, we use the same path for all values of \(\rho\), in order to evaluate clearly the influence of \(\rho\). Our results are represented in Figure 4, which shows that the total energy dissipated by JPEG in the whole path is high compared to the energy dissipated with T-JPEG for each \(\rho\) equal to 2, 4, and 6, which confirms that T-JPEG consumes less energy compared to JPEG, especially when \(\rho\) is relatively small.

\(^1\)The cycle starts by image capturing and ends at the reception of the image by the sink.

To show the effect of \(\rho\) on the cluster lifetime, we consider the following scenario. The same node from one cluster is taking, processing and sending iteratively the same image using one selected \(\rho\) until the total depletion of its energy. After that, a second node from the same cluster does the same task (i.e., for the same image and with the same \(\rho\)) until no node inside this cluster has enough energy to process and send the image. If we consider the same example as before (Lena), we obtain Figure 5 which confirms that if the selected \(\rho\) increases for many cycles, the source node die quickly, and the cluster lifetime becomes shorter as the number of nodes inside this cluster decreases. It is clear from Figure 5 that T-JPEG performs better than JPEG for various value of \(\rho\). Starting with 15 nodes inside one cluster, with JPEG the cluster lifetime is limited to 34 cycles, while with T-JPEG, for \(\rho = 6\) and 4, the cluster lifetime is prolonged to 78 and 112 cycles, respectively. And for \(\rho = 1\), there remains 10 nodes after 120 cycles. This simulation confirms again, that T-JPEG improves energetically JPEG.

VII. CONCLUSION

We studied the problem of modeling and adapting JPEG to the energy requirements of VSN. First, we introduced a new energy model for JPEG compression stages. This model is very simple since it is based only on the basic operations needed at each stage of JPEG, and it does not take into account the complexity of implementation. This model may serve as a mean to simulate and evaluate the energy consumed by a given visual sensor when executing JPEG. Finally, we suggested an adapted version of JPEG, called T-JPEG. With T-JPEG we processed only the upper-left triangular portion of each block of \(k \times k\) DCT coefficients of a given image. Our simulations show how T-JPEG improves energetically JPEG, while maintaining an adequate image quality at the sink. One direction for our future work is to study the effects of varying \(\rho\) on other performance metrics like the processing time, the bandwidth and the network state.

REFERENCES