

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

وزارة التعليم العالي والبحث العلمي

BADJI MOKHTAR – ANNABA UNIVERSITY  
UNIVERSITE BADJI MOKHTAR – ANNABA



جامعة باجي مختار – عنابة

Faculté des Sciences de l'Ingénierat

Année : 2019/2020

Département d'Informatique

## THÈSE

Présentée en vue de l'obtention du diplôme de

**Doctorat ès Sciences**

---

# Compression d'images sous la contrainte de l'énergie appliquée aux réseaux de capteurs sans fil

---

**Filière** : Informatique

**Option** : Informatique Embarquée (INEM)

par **ARAAR Chaouki**

DEVANT LE JURY

**Président** : M. BOUDOUR Rachid

Professeur à l'Université d'Annaba

**Directeur** : M. GHANEMI Salim

Professeur à l'Université d'Annaba

**Co-directeur** : M. BENMOHAMMED Mohammed

Professeur à l'Université Constantine 2

**Examineurs** : M. CHIKHI Salim

Professeur à l'Université Constantine 2

M. FARAH Nadir

Professeur à l'Université d'Annaba

M. SERIDI Hamid

Professeur à l'Université de Guelma



*Dédicace*

*À mes chers parents;*

*À ma femme;*

*À mon fils Mohammed Bara;*

*À mes frères et ma sœur;*

*À mes belles-sœurs;*

*À ma nièce Rafif et mon petit neveu Ahmed-Siraj;*

*À toute ma famille;*

*À tous mes amis;*

*À tous ceux qui me sont chers;*

# Remerciements

الحمد لله الذي بنعمته تتم الصالحات، ويتوفيقه تتحقق المقاصد و الغايات.

Mes remerciements les plus sincères et les plus chaleureuses s'adressent en premier lieu à mes directeurs de thèse monsieur GHANEMI Salim, Professeur à l'Université Badji Mokhtar – Annaba et monsieur BENMOHAMMED Mohammed, Professeur à l'Université Abdelhamid Mehri – Constantine 2, pour m'avoir confié ce travail de recherche, pour leur gentillesse, leur disponibilité ainsi que leurs précieux conseils tout au long de ces années de doctorat.

Je tiens à exprimer toute ma reconnaissance et mon profond respect à monsieur BOUDOUR Rachid, Professeur à l'Université Badji Mokhtar – Annaba, qui m'a fait l'honneur de présider le jury de thèse.

J'adresse aussi mes sincères remerciements aux membres de jury, monsieur CHIKHI Salim, Professeur à l'Université Abdelhamid Mehri – Constantine 2, monsieur FARAH Nadir, Professeur à l'Université Badji Mokhtar – Annaba et monsieur SERIDI Hamid, Professeur à l'université de Guelma d'avoir bien voulu examiner et juger ce travail.

Je témoigne ici ma profonde gratitude et mes vifs remerciements à monsieur KOUADRIA Nasreddine et monsieur ATOUI Hamza pour leurs aides précieuses.

## الملخص

تعتبر الأعمال المقدمة في هذه الأطروحة جزءاً من تطوير طرق ضغط الصور مع الأخذ بعين الاعتبار كفاءة استخدام الطاقة في سياق شبكات الاستشعار اللاسلكية. الهدف الرئيسي هو تقليل استهلاك الطاقة من أجهزة الاستشعار، وبالتالي، الحفاظ على عمر طويل للشبكة.

تتعلق مساهمة هذه الأطروحة بشكل رئيسي في التقليل من التعقيد الخوارزمي لنظام ضغط الصور JPEG. هذه الخوارزمية جد مستهلكة للوقت والطاقة بسبب تعقيدها العالي، وبشكل أدق، بسبب تعقيد مرحلة تحويل جيب التمام المتقطع (دي سي تي). من أجل تكييف هذه الخوارزمية مع القيود الخاصة لشبكات الاستشعار اللاسلكية، اقترحنا تقليل التعقيد الحسابي لدي سي تي من خلال الجمع بين طريقة دي سي تي التقريبية مع تقنية التقييم. الهدف من الأولى هو تقليل التعقيد الحسابي باستخدام عمليات الجمع فقط بدلا من عمليات الضرب المكلفة، بينما تهدف التقنية الثانية إلى حساب معاملات التردد المنخفض الأكثر أهمية فقط. تم تطوير خوارزمية للحساب السريع للتقنية المقترحة. فقط عشر عمليات جمع تكفي لحساب التحويل المباشر و العكسي. يتميز التحويل المقترح بتعقيد حسابي منخفض للغاية مع الحفاظ على أداء تنافسي لضغط الصور مقارنةً بأحدث الأساليب. تم تنفيذ تجارب محاكاة التطبيق البرمجي لإثبات كفاءة اقتراحنا من حيث سرعة وقت التنفيذ، الذاكرة المطلوبة واستهلاك الطاقة مقارنةً بأحدث الأساليب. نفذت الأعمال التجريبية باستخدام المتحكم الدقيق Atmel ATmega128 للوحات أجهزة الاستشعار Mica2 و MicaZ. كما تم تصميم هندسة متوازية-متوالية فعالة لدي سي تي المقترحة. نفذ التصميم الناتج على مصفوفة البوابات المنطقية القابلة للبرمجة Xilinx Virtex-6 XC6VSX475T-2ff1156 و تم تقييمها لاستخدام موارد الأجهزة، استهلاك الطاقة وكفاءة الأداء في الوقت الفعلي. أظهرت جميع المقاييس التي درسناها مزايا واضحة للتصميم المقترح مقارنةً بأحدث الأساليب المنافسة.

**الكلمات المفتاحية :** شبكات الاستشعار اللاسلكية، شبكات الاستشعار اللاسلكية البصرية، ضغط الصور، دي سي تي التقريبية، تقنية التقييم، هندسة دائرة التكامل الفائق، مصفوفة البوابات المنطقية القابلة للبرمجة، حفظ الطاقة.

## Abstract

The works presented in this thesis are part of the development of energy-aware image compression methods in the context of wireless sensor networks. The main objective is to decrease the energy consumption of sensors and thus, to maintain a long network lifetime.

The contribution of this thesis focuses mainly on the reduction of the algorithmic complexity of the JPEG image compression standard. This algorithm is very greedy in time and energy because of its high complexity, and more precisely, due to the complexity of its discrete cosine transform (DCT) stage. In order to adapt this algorithm to the particular constraints of wireless sensor networks, we proposed to reduce the computational complexity of the DCT by combining an approximate DCT method with a pruning approach. The aim of the former is to reduce the computational complexity by using only additions instead of costly multiplication operations, while the latter aims at computing only the more important low-frequency coefficients. An algorithm for the fast computation of the proposed transform is developed. Only ten additions are required for both forward and backward transformations. The proposed pruned DCT transform exhibits extremely low computational complexity while maintaining competitive image compression performance in comparison with the state-of-the-art methods. Simulation experiments of the software-based implementation are provided to prove the efficiency of the proposal in terms of processing time, required memory and energy consumption in comparison with state-of-the-art methods. Experimental works are conducted using the Atmel ATmega128 processor of Mica2 and MicaZ sensor boards. Efficient parallel-pipelined hardware architecture for the proposed pruned DCT is also designed. The resulting design is implemented on Xilinx Virtex-6 XC6VSX475T-2ff1156 FPGA technology and evaluated for hardware resource utilization, power consumption and real-time performance. All the metrics we investigated showed clear advantages of the proposed design over the state-of-the-art competitors.

**Keywords:** Wireless sensors networks, visual sensors networks, image compression, DCT approximation, pruning approach, VLSI architecture, FPGA, energy conservation

## Résumé

Les travaux présentés dans cette thèse s'inscrivent dans le cadre de développement de méthodes de compression d'images tenant compte de l'efficacité énergétique dans le contexte des réseaux de capteur sans fil. L'objectif principal est de réduire la consommation d'énergie des nœud-capteurs et donc, de maintenir une longue durée de vie du réseau.

La contribution de cette thèse porte principalement sur la réduction de la complexité algorithmique du standard de compression d'images JPEG. Cet algorithme est très gourmand en temps et en énergie à cause de sa complexité élevée, et plus précisément, en raison de la complexité de son étape de transformation en cosinus discrète (DCT). Afin d'adapter cet algorithme aux contraintes particulières des réseaux de capteurs sans fil, nous avons proposé de réduire la complexité calculatoire de la DCT en combinant une méthode de DCT approximative avec une approche de sélection zonale des coefficients. Le but de la première est de réduire le coût de calcul de la DCT en n'employant que des opérations d'addition au lieu des opérations coûteuses de multiplication, tandis que la seconde vise à calculer uniquement les coefficients de basse fréquence les plus importants. Un algorithme pour le calcul rapide de la DCT proposée est développé. Seulement dix opérations d'addition sont requises pour le calcul de la transformation directe et inverse. La DCT proposée présente une complexité de calcul extrêmement faible tout en maintenant des performances de compression d'images compétitives par rapport aux méthodes de l'état de l'art. Des expériences de simulation de l'implémentation logicielle sont fournies en vue de prouver l'efficacité de notre proposition en termes de temps d'exécution, mémoire requise et consommation d'énergie en comparaison avec les méthodes de l'état de l'art. Les travaux expérimentaux ont été effectués en considérant le microcontrôleur Atmel ATmega128 embaqué sur le nœud-capteur Mica2 et MicaZ. Une architecture matérielle parallèle-pipelinée pour le calcul de la DCT proposée est également conçue. Cette architecture est implémentée sur un circuit FPGA Xilinx Virtex-6 XC6VSX475T-2ff1156 et évaluée pour l'utilisation des ressources matérielles, consommation d'énergie et performance temps réel. Toutes les mesures que nous avons étudiées ont montré des avantages évidents du design proposé par rapport aux concurrents de l'état de l'art.

**Mots-clés :** Réseaux de capteurs sans fil, réseaux de capteurs d'images, compression d'images, DCT approximative, approche zonale, architecture VLSI, FPGA, conservation d'énergie.

## Table des matières

Dédicace .....	i
Remerciements.....	ii
المخلص .....	iii
Abstract.....	iv
Résumé.....	v
Table des matières.....	vi
Liste des figures .....	ix
Liste des tableaux.....	xi
Liste des acronymes .....	xii
Introduction générale .....	xv
1. Contexte général et problématique.....	1
2. Contributions de la thèse .....	3
3. Organisation du manuscrit .....	3

### **I. Positionnement : Etat de l'art sur la compression d'images dans les réseaux de capteurs sans fil**

#### **Chapitre 1 : Introduction aux réseaux de capteurs sans fil**

1.1 Introduction.....	5
1.2 Architecture de réseau de capteurs .....	6
1.2.1 Architecture matérielle d'un nœud-capteur.....	6
1.2.2 Architecture de communication.....	7
1.2.3 Architecture protocolaire .....	8
1.3 Contraintes de conception des réseaux de capteurs .....	9
1.3.1 Tolérance aux pannes .....	10
1.3.2 Passage à l'échelle .....	10
1.3.3 Coût de fabrication .....	10
1.3.4 Environnement d'exploitation .....	10
1.3.5 Topologie de réseau.....	11
1.3.6 Contraintes matérielles .....	11
1.3.7 Support de transmission.....	11
1.3.8 Consommation d'énergie.....	12

1.4	Classification des réseaux de capteurs .....	12
1.4.1	Modèle de collecte de données .....	12
1.4.2	Modèle de communication .....	13
1.4.3	Modèle de mobilité .....	15
1.5	Consommation d'énergie d'un nœud-capteur .....	16
1.5.1	Formes de dissipation d'énergie .....	16
1.5.2	Sources de surconsommation d'énergie .....	17
1.6	Techniques de conservation de l'énergie .....	19
1.6.1	Notion de la durée de vie d'un réseau de capteurs .....	19
1.6.2	Taxonomie des approches de conservation d'énergie .....	21
1.7	Réseau de capteurs d'images .....	24
1.7.1	Spécificités des réseaux de capteurs d'images .....	25
1.7.2	Plateformes de capture d'image .....	26
1.7.3	Exemples d'application .....	30
1.8	Conclusion .....	33

## **Chapitre 2 : Compression d'images à la source dans les réseaux de capteurs sans fil**

2.1	Introduction.....	35
2.2	Classification des techniques de compression d'images .....	35
2.3	Algorithmes de compression sans perte.....	36
2.3.1	Dé-corrélation.....	36
2.3.2	Codage entropique.....	37
2.4	Algorithmes de compression avec pertes.....	38
2.4.1	Algorithmes non-basés sur une transformée .....	38
2.4.2	Algorithmes à base de transformée .....	39
2.5	Conclusion .....	52

## **II. Contributions : Vers la compression d'images efficace en énergie dans les réseaux de capteurs sans fil**

### **Chapitre 3 : Propriétés et approches de réduction de la DCT**

3.1	Introduction.....	53
3.2	Transformée en cosinus discrète.....	54
3.2.1	DCT unidimensionnelle.....	54

3.2.2	DCT bidimensionnelle.....	58
3.3	Propriétés de la DCT 2-D .....	62
3.3.1	Dé-corrélation.....	63
3.3.2	Compactage d'énergie .....	65
3.3.3	Séparabilité .....	65
3.3.4	Symétrie.....	67
3.3.5	Orthogonalité .....	68
3.3.6	Normalisation .....	68
3.3.7	Orthonormalité.....	69
3.4	Approche de réduction de la DCT .....	69
3.4.1	DCT rapide .....	69
3.4.2	DCT zonale.....	75
3.5	Conclusion .....	76
 <b>Chapitre 4 : DCT zonale approximative et architecture matérielle proposées</b>		
4.1	Introduction.....	78
4.2	DCT zonale exacte et approximative.....	79
4.3	DCT zonale approximative proposée.....	81
4.3.1	Transformée proposée .....	81
4.3.2	Évaluation de la complexité .....	84
4.4	Évaluation des performances .....	86
4.4.1	Temps d'exécution et consommation d'énergie .....	86
4.4.2	Mémoire requise .....	88
4.4.3	Compression d'images .....	89
4.5	Architecture VLSI.....	93
4.5.1	Architecture matérielle proposée.....	94
4.5.2	Implémentation sur FPGA.....	97
4.6	Conclusion .....	98
Conclusions et perspectives .....		100
1.	Conclusions .....	100
2.	Perspectives.....	101
Liste des publications.....		103
Références.....		104

## Liste des figures

<b>Figure 1.1.</b> Schéma conventionnel d'un réseau de capteurs sans fil.....	5
<b>Figure 1.2.</b> Anatomie générale d'un nœud capteur.....	6
<b>Figure 1.3.</b> Architecture de communication d'un RCSF.....	8
<b>Figure 1.4.</b> Pile protocolaire des réseaux de capteurs sans fil [Wang 2001]. .....	9
<b>Figure 1.5.</b> Déférents scénarios d'applications pour réseaux de capteurs sans fil.....	14
<b>Figure 1.6.</b> Modèle de communication dans un réseau de capteurs.....	15
<b>Figure 1.7.</b> Consommation d'énergie du nœud-capteur MicaZ [Akyildiz 2010]. .....	18
<b>Figure 1.8.</b> Taxonomie des approches de conservation d'énergie dans les RCSFs.....	22
<b>Figure 1.9.</b> Exemple d'un réseau de capteurs d'images.....	25
<b>Figure 1.10.</b> Caméra Cyclops attachée à un mote Mica2.....	27
<b>Figure 1.11.</b> CMUCam3 attachée à un mote Tmote Sky.....	28
<b>Figure 1.12.</b> Plateforme basée sur iMote2.....	28
<b>Figure 1.13.</b> Caméra CITRIC avec un mote TelosB.....	29
<b>Figure 1.14.</b> Un imageur ALOHA attachée à un mote Mica2.....	30
<b>Figure 1.15.</b> Projet <i>BearCam</i> pour l'observation des ours bruns « grizzly ».....	31
<b>Figure 1.16.</b> Image d'une lecture d'un compteur d'eau.....	33
<b>Figure 2.1.</b> Classification des algorithmes de compression d'images.....	37
<b>Figure 2.2.</b> Schéma fonctionnel d'un quantificateur vectoriel.....	38
<b>Figure 2.3.</b> Schéma générale de compression/décompression à base de transformée.....	40
<b>Figure 2.4.</b> Étapes de la compression JPEG.....	44
<b>Figure 2.5.</b> Décomposition en ondelettes à deux niveaux de l'image Barbera.....	47
<b>Figure 2.6.</b> Les sous-bandes de la transformée en ondelettes et leurs contenus.....	47
<b>Figure 2.7.</b> Chaîne de compression JPEG2000.....	48
<b>Figure 2.8.</b> Etapes de la compression SPIHT.....	50
<b>Figure 3.1.</b> Fonctions de base de la DCT 1-D ( $N = 8$ ).....	57
<b>Figure 3.2.</b> Un bloc d'échantillons $8 \times 8$ de l'image Lena.....	60
<b>Figure 3.3.</b> Coefficients DCT pour le bloc $8 \times 8$ de l'image Lena.....	60
<b>Figure 3.4.</b> Répartition des fréquences dans un bloc DCT $8 \times 8$ .....	60
<b>Figure 3.5.</b> Fonctions (sous-images) de base de la DCT 2-D ( $N = 8$ ). Le gris neutre représente zéro, le blanc représente les amplitudes positives et le noir les amplitude négative [Pennebaker 1993].....	61

<b>Figure 3.6.</b> Exemple d'une image en niveau de gris typique (Boat).....	63
<b>Figure 3.7.</b> Diagramme de distribution des paires de valeurs de pixels adjacents. (a) avant la DCT. (b) après la DCT.....	64
<b>Figure 3.8.</b> (a) Baboon et sa DCT. (b) Child et sa DCT. (c) Saturn et sa DCT. ....	66
<b>Figure 3.9.</b> Calcul de la DCT 2-D en utilisant deux transformations successives de DCT 1-D. ....	67
<b>Figure 3.10.</b> Approches de réduction de la DCT. ....	70
<b>Figure 3.11.</b> Principe de la DCT zonale ( $k = 4$ ). (a) forme carrée. (b) forme triangulaire.....	76
<b>Figure 4.1.</b> Le graphe de flux de la matrice $T_4$ proposée. Les flèches en pointillés signifient des multiplications par $-1$ .....	84
<b>Figure 4.2.</b> DCT 2-D zonale approximative $4 \times 4$ proposée ( $k = 4$ ). ....	84
<b>Figure 4.3.</b> Image de test Peppers encodée à 0,5 bpp. ....	91
<b>Figure 4.4.</b> Image de test Boat encodée à 0,5 bpp... ..	92
<b>Figure 4.5.</b> Performances débit-distorsion. (a) PSNR moyenne. (b) SSIM moyenne. ....	93
<b>Figure 4.6.</b> Architecture parallèle-pipelinée proposée pour le calcul de la DCT 2-D approximative. ....	94
<b>Figure 4.7.</b> Architecture parallèle-pipelinée proposée pour la DCT 1-D approximative à 8-points. (a) DCT améliorée [Potluri 2014]. (b) DCT proposée. ....	95
<b>Figure 4.8.</b> Circuit détaillé du tampon de transposition. (a) DCT ordinaire (b) DCT zonale. ....	96

---

## Liste des tableaux

<b>Table 3.1.</b> Complexité de calcul des algorithmes rapides de la DCT 1-D et 2-D.....	71
<b>Table 4.1.</b> Evaluation de la complexité arithmétique des algorithmes de calcul de la DCT pour un bloc $8 \times 8$ .....	86
<b>Table 4.2.</b> Temps d'exécution, énergie consommée, ETIR et taille du code d'un bloc $8 \times 8$ obtenus par le simulateur Atmel AVR studio 4 sur le microcontrôleur Atmel ATmega128. .	87
<b>Table 4.3.</b> Mesures du PSNR et SSIM moyennes données à 0,5 bpp.....	90
<b>Table 4.4.</b> Utilisation des ressources matérielles, vitesse d'horloge et consommation d'énergie de différents designs de DCT 2-D sur un FPGA Xilinx Virtex-6 XC6V SX475T-2ff1156.....	98

## Liste des acronymes

<b>AC</b>	Alternative Current
<b>ARM</b>	Advanced RISC Machine
<b>AT</b>	Area Time
<b>BAS</b>	Bouguezel Ahmad Swamy
<b>bpp</b>	bit per pixel
<b>CAN</b>	Convertisseur Analogique Numérique
<b>CENS</b>	Center for Embedded Network Sensing
<b>CIF</b>	Common Interchange Format
<b>CLB</b>	Configurable Logic Block
<b>CL-DCT</b>	Cordic Loeffler Discrete Cosine Transform
<b>CMOS</b>	Complementary Metal Oxide Semiconductor
<b>CORDIC</b>	Coordinate Rotation Digital Computer
<b>COTS</b>	Commercial Off The Shelf
<b>CPLD</b>	Complex Programmable Logic Device
<b>DA</b>	Distributed Arithmetic
<b>dB</b>	Decibels
<b>DC</b>	Direct Current
<b>DCT</b>	Discrete Cosine Transform
<b>DCT 1-D</b>	Discrete Cosine Transform one Dimension
<b>DCT 2-D</b>	Discrete Cosine Transform two Dimension
<b>DFT</b>	Discrete Fourier Transform
<b>DST</b>	Discrete Sine Transform
<b>DWT</b>	Discrete Wavelet Transform
<b>EBCOT</b>	Embedded Block Coding with Optimized Truncation
<b>EPE</b>	Energy Packing Efficiency
<b>ETIR</b>	Execution Time Improvement Ratio
<b>EZW</b>	Embedded Zero-tree Wavelet
<b>FF</b>	Flip Flop
<b>FPGA</b>	Field Programmable Gate Array
<b>FPS</b>	Frames per Second
<b>Gbit/s</b>	Gigabit/second

<b>HVS</b>	Human Visual System
<b>ISM</b>	Industriel Scientific and Medical
<b>JPEG</b>	Joint Photographic Expert Group
<b>JPEG2000</b>	Joint Photographic Experts Group 2000
<b>KLT</b>	Karhunen Loeve Transform
<b>LLM</b>	Loeffler Ligtenberg Moschytz
<b>LUT</b>	Look-up Table
<b>LZW</b>	Lempel Ziv Welch
<b>MAC</b>	Medium Access Control
<b>MEMS</b>	Micro Electro Mechanical Systems
<b>MHz</b>	Megahertz
<b>MIPS</b>	Millions of Instructions per Second
<b>MMC/SD</b>	Multimedia Card/Smart Digital
<b>MRDCT</b>	Modified Rounded Discrete Cosine Transform
<b>MSE</b>	Mean Square Error
<b>MTE</b>	Minimize Total Energy
<b>mW</b>	milliWatt
<b>NEDA</b>	New Distributed Arithmetic
<b>ns</b>	nanosecond
<b>OSI</b>	Open System Interconnection
<b>PAR</b>	Place and Routing
<b>PSNR</b>	Peak Signal to Noise Ratio
<b>QCIF</b>	Quarter Common Interchange Format
<b>QoS</b>	Quality of Service
<b>QV</b>	Quantification Vectorielle
<b>QVGA</b>	Quarter Video Graphics Array
<b>RAM</b>	Random Access Memory
<b>RCSF</b>	Réseau de Capteurs Sans Fil
<b>RDCT</b>	Rounded Discrete Cosine Transform
<b>RGB</b>	Red Green Blue
<b>RISC</b>	Reduced Instruction Set Computer
<b>RLC</b>	Run Length Coding
<b>RLE</b>	Run Length Encoding

<b>ROM</b>	Read Only Memory
<b>RX</b>	Receive
<b>SDCT</b>	Signed Discrete Cosine Transform
<b>SPIHT</b>	Set Partitioning in Hierarchical Trees
<b>SRAM</b>	Static Random Access Memory
<b>SSIM</b>	Structural SIMilarity
<b>TX</b>	Transmit
<b>UCLA</b>	Université de Californie à Los Angeles
<b>VGA</b>	Video Graphics Array
<b>VLSI</b>	Very Large Scale Integration



# Introduction générale

## Introduction générale

### Sommaire

---

1.	Contexte général et problématique.....	1
2.	Contributions de la thèse.....	3
3.	Organisation du manuscrit .....	3

---

## **1. Contexte général et problématique**

Les progrès récents dans les technologies de communication sans fil, conjointement avec les développements de la microélectronique ont mis à disposition un nouveau type de réseau de communication constitué de capteurs sans fil, très petits, peu onéreux et alimentés par batterie. Les réseaux de capteur sans fil, comme ils sont nommés, sont des réseaux sans fil auto-configurés et sans infrastructure, constitués de petits dispositifs dotés de capacités de calcul, de stockage, de communication et surtout d'énergie très limitées. L'objectif principal de ces réseaux est de collecter des données de l'environnement et de les envoyer à une station collectrice où les données peuvent être observées et analysées [[Akyildiz 2002](#)].

Les réseaux de capteurs sans fil ont attiré un large éventail de disciplines où des proches interactions avec le monde physique sont essentielles. Les capacités de détection distribuée et la facilité de déploiement fournies par un paradigme de communication sans fil font des réseaux de capteurs une composante importante de notre vie quotidienne. En conséquence, une large gamme d'applications peut être envisagée pour de tels réseaux. Citons à titre d'exemples, la surveillance de l'environnement, la détection de l'habitat, l'agriculture de précision, la gestion des catastrophes, la surveillance de la santé et les applications militaires, ...etc.

La conception et l'implémentation des réseaux de capteur sans fil font face à plusieurs défis, principalement en raison de la contrainte de l'énergie. C'est pour cette raison que la plupart des recherches sur les réseaux de capteur se sont concentrées sur la conception d'algorithmes et de méthodes pour le traitement et la communication de données efficaces en énergie. L'efficacité énergétique est bien sûr une préoccupation fondamentale pour obtenir la durée de vie du réseau requise, car le remplacement des batteries des nœuds-capteurs est souvent difficile, voire impossible.

Assez récemment, l'avènement de capteurs d'images en technologie CMOS à faible coût et à faible consommation, tel que la caméra Cyclops [[Rahimi 2005](#)], a conduit à l'émergence d'un nouveau paradigme, appelé réseaux de capteurs d'images [[Akyildiz 2007](#), [Soro 2009](#)]. Dans de tels réseaux, chaque nœud-capteur est chargé d'acquérir, de traiter et de transmettre des informations visuelles à la station de base via le chemin sans fil établi entre la source et la destination. Malheureusement, la transmission de données visuelles comme par exemple une image, nécessite plus de consommation d'énergie et plus d'utilisation de la

bande passante que la transmission de données scalaires simples, outre les coûts additionnels pour l'acquisition, le traitement et le stockage de données au niveau du nœud-capteur. En effet, le volume associé à une image est supérieur à celui réservé à une simple donnée scalaire, telle que la température ou l'humidité. Contrairement aux données scalaires qui peuvent être codées sur quelques octets (par exemple, pour une mesure de température, 2 ou 3 octets sont plus que suffisants) et, donc elles peuvent être transportées sur un seul paquet de données, une image est généralement représentée sur plusieurs milliers d'octets selon sa taille et sa résolution. Par conséquent, le nœud-capteur engagera plusieurs paquets afin de pouvoir transmettre l'image entière et, donc beaucoup d'énergie sera dissipée par le nœud source lui-même ainsi que les nœuds intermédiaires transmettant les paquets de données vers la station collectrice.

La plupart des études sur les réseaux manipulant des données scalaires ont typiquement assumées que les coûts de calcul, comprenant l'acquisition, prétraitement et la compression sont négligeable comparés aux coûts de la communication [[Krishnamachari 2002](#)]. Cette hypothèse peut convenir aux capteurs de données scalaires manipulant des paramètres physiques légers, telle que la température ou l'humidité, pour lesquels le coût de la compression des données, si elle est effectuée, est négligeable par rapport au coût de la communication. Dans le cas des réseaux de capteurs d'images qui produisent une pléthore de flux redondants, cette hypothèse n'est plus remplie, car la grande quantité de flux visuels nécessite toujours de la compression [[Mammeri 2012](#)]. Mais, la compression d'images dans ce contexte est-elle vraiment efficace en énergie?

Bien que les algorithmes classiques de compression d'images existants, tels que SPIHT, JPEG2000 ou encore JPEG, ont fourni d'excellentes performances en termes de rapport débit-distorsion, ils ne tiennent pas compte des contraintes imposées par les réseaux de capteurs. En fait, tous ces algorithmes exigent beaucoup de calcul et d'accès mémoire, tandis que la plupart des plates-formes de nœuds-capteurs sont des systèmes limitées en ressources et n'ont pas suffisamment de mémoire, de capacités de traitement et d'énergie pour effectuer des tâches de compression complexes. Des travaux comme ceux de Ferrigno et al. [[Ferrigno 2005](#)] ont montré que des algorithmes standards bien connus sont très gourmands en temps et en énergie, ce qui signifie que l'opération de compression consomme plus d'énergie qu'une transmission de l'image brute sans compression. Par conséquent, de nouvelles stratégies sont nécessaires pour construire de nouveaux schémas de compression satisfaisant les contraintes

inhérentes aux réseaux de capteurs, car l'adoption des méthodes conventionnelles n'est pas toujours très bénéfique en termes de consommation d'énergie.

## **2. Contributions de la thèse**

Cette thèse constitue une contribution au développement de nouvelles méthodes de compression d'images tenant compte de l'efficacité énergétique, afin de prolonger la durée de vie du nœud-capteur et ainsi, la durée de vie du réseau tout entier. Notre contribution propose d'adapter l'algorithme bien connu JPEG aux exigences des réseaux de capteurs sans fil. Cet algorithme s'appuie, généralement, sur une étape de dé-corrélation des données basée sur une transformée en cosinus discrète (DCT), qui est la plus exigeante en matière d'énergie [Taylor 2002]. Notre objectif dans la présente thèse est de réduire la complexité de calcul de cette transformée en introduisant une nouvelle DCT zonale approximative, autrement dit, d'utiliser une approche de sélection zonale de coefficients conjointement avec une approche de DCT approximative. Cela permet de réduire considérablement la complexité de calcul de la DCT et, par conséquent, la consommation d'énergie du nœud-capteur. De plus, nous nous intéressons à la conception d'une architecture matérielle efficace et peu complexe, qui contribue à soulager le microcontrôleur du nœud-capteur du coût de calcul de la DCT et qui permet de réaliser un traitement à haute vitesse et à faible consommation d'énergie.

## **3. Organisation du manuscrit**

Ce manuscrit est organisé en quatre chapitres répartis en deux parties suivi d'une conclusion générale et perspectives. La partie I composée des chapitres 1 et 2, constitue un état de l'art sur la compression d'images à la source dans les réseaux de capteurs sans fil.

Le chapitre 1 donne un aperçu de l'état de l'art sur les réseaux de capteurs sans fil en présentant l'architecture matérielle, protocolaire et de communication ; les contraintes de conception et les modèles de fonctionnement de cette technologie importante. Puis, nous recensons les différents facteurs provoquant la dissipation d'énergie d'un nœud-capteur et discutons brièvement les principales techniques de conservation d'énergie trouvées dans la littérature. Ce chapitre présente également le cas spécifique des réseaux de capteurs d'images, en évoquant leurs disparités par rapport aux réseaux de capteurs de données scalaires, les technologies matérielles de capture d'images utilisées dans ces réseaux ainsi que leurs applications potentielles.

Le chapitre 2, présente une revue de la littérature des techniques de compression d'images dans le domaine générale de la compression d'images. Deux paradigmes majeurs seront considérés, à savoir la compression sans perte et celle avec pertes. Il décrit brièvement les idées de base derrière ces approches, leurs avantages et inconvénients, ainsi que leurs applications dans les réseaux de capteurs sans fil.

La partie II composée des chapitres 3 et 4, est d'une part, un passage en revue de la transformée en cosinus discrète (DCT) et aux approches de réduction de sa complexité et, d'autre part, elle détaille notre contribution et les résultats de l'évaluation des performances de son implémentation.

Le chapitre 3 est centré sur la présentation de la DCT. Son objectif est double. Il s'agit premièrement, de présenter les aspects mathématiques de base de la DCT ainsi que les propriétés les plus importantes et utiles de cette transformation, et deuxièmement, de discuter les différentes techniques archivées dans la littérature servant à réduire sa complexité calculatoire.

Le chapitre 4 est tout entier consacré à notre contribution. Il introduit d'abord la DCT zonale approximative de faible complexité proposée pour la compression d'images dans les réseaux de capteurs sans fil qui, par la suite sera évaluée en termes de temps d'exécution, mémoire requise, consommation d'énergie et qualité d'image par rapport aux méthodes de l'état de l'art. Après cela, le chapitre traite l'architecture matérielle parallèle-pipelinée conçue pour la DCT zonale proposée et les méthodes concurrentes. Il se termine par l'évaluation des performances du design proposé comparé aux concurrents de l'état de l'art lorsqu'ils sont implémentés sur circuit logique programmable suivant la technologie FPGA.

## Première partie

# **I. Positionnement : Etat de l'art sur la compression d'images dans les réseaux de capteurs sans fil**



# Chapitre 1

## Chapitre 1 : Introduction aux réseaux de capteurs sans fil

### Sommaire

---

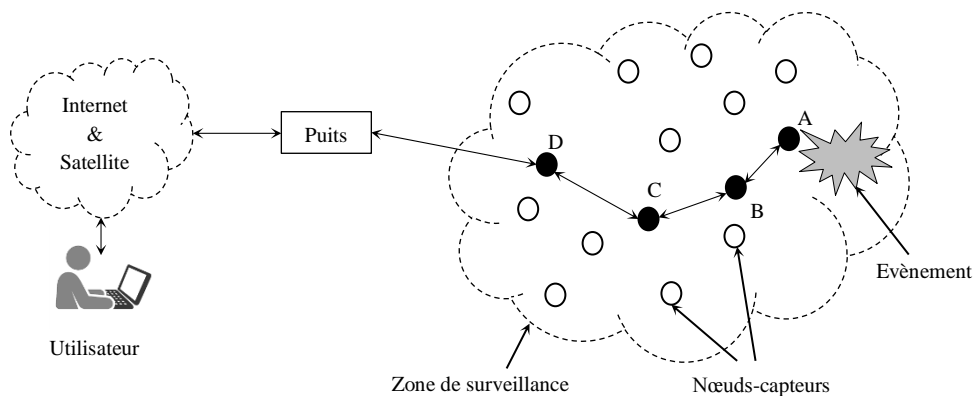
<b>1.1</b>	<b>Introduction</b> .....	<b>5</b>
<b>1.2</b>	<b>Architecture de réseau de capteurs</b> .....	<b>6</b>
1.2.1	Architecture matérielle d'un nœud-capteur .....	6
1.2.2	Architecture de communication .....	7
1.2.3	Architecture protocolaire .....	8
<b>1.3</b>	<b>Contraintes de conception des réseaux de capteurs</b> .....	<b>9</b>
1.3.1	Tolérance aux pannes.....	10
1.3.2	Passage à l'échelle.....	10
1.3.3	Coût de fabrication .....	10
1.3.4	Environnement d'exploitation.....	10
1.3.5	Topologie de réseau .....	11
1.3.6	Contraintes matérielles.....	11
1.3.7	Support de transmission.....	11
1.3.8	Consommation d'énergie.....	12
<b>1.4</b>	<b>Classification des réseaux de capteurs</b> .....	<b>12</b>
1.4.1	Modèle de collecte de données.....	12
1.4.2	Modèle de communication .....	13
1.4.3	Modèle de mobilité.....	15
<b>1.5</b>	<b>Consommation d'énergie d'un nœud-capteur</b> .....	<b>16</b>
1.5.1	Formes de dissipation d'énergie.....	16
1.5.2	Sources de surconsommation d'énergie .....	17
<b>1.6</b>	<b>Techniques de conservation de l'énergie</b> .....	<b>19</b>
1.6.1	Notion de la durée de vie d'un réseau de capteurs .....	19
1.6.2	Taxonomie des approches de conservation d'énergie .....	21
<b>1.7</b>	<b>Réseau de capteurs d'images</b> .....	<b>24</b>
1.7.1	Spécificités des réseaux de capteurs d'images.....	25
1.7.2	Plateformes de capture d'images .....	26
1.7.3	Exemples d'application.....	30
<b>1.8</b>	<b>Conclusion</b> .....	<b>33</b>

---

## 1.1 Introduction

Les progrès récents réalisés dans les systèmes micro électromécaniques et les technologies de communication sans fil ont permis la production à faible coût de dispositifs de détection minuscules, alimentés par batterie et dotés de capacités de traitement, de stockage et de communication limitées. Ces petits dispositifs de détection appelés communément nœuds-capteurs sont déployés de manière dense dans une région d'intérêt à grande échelle afin de collecter collectivement des données environnementales ou physiques et de les transmettre à une station de base. Les nœuds-capteurs travaillent de manière coopérative pour former un réseau de capteurs sans fil (RCSF) auto-organisé, auto-configurable et sans infrastructure (ad-hoc) [Akyildiz 2002]. Bien entendu, la conservation de l'énergie est l'une des préoccupations majeures pour le maintien du réseau en fonctionnement, car la batterie des nœuds-capteurs est en général de capacité limitée et non rechargeable.

Comme illustré sur la figure 1.1, un réseau de capteur sans fil est un réseau ad hoc composé d'un grand nombre de nœuds-capteurs qui sont, soit posés à des emplacements préétablis, soit dispersés aléatoirement (souvent largués par voie aérienne à l'aide d'avions ou hélicoptères) dans une zone géographique, appelée zone de surveillance. Par principe, les nœuds-capteurs remplissent deux rôles : un rôle de source d'informations et un rôle de relais pour le reste du réseau. Le point de collecte, appelé puits ou station de base, agit en tant que passerelle pour le réseau de capteurs. Un réseau à infrastructure (internet et satellite) permet la connexion à distance d'un utilisateur éloigné [Akyildiz 2002].



**Figure 1.1.** Schéma conventionnel d'un réseau de capteurs sans fil.

## 1.2 Architecture de réseau de capteurs

### 1.2.1 Architecture matérielle d'un nœud-capteur

Le nœud-capteur sans fil constitue l'élément central d'un réseau de capteurs sans fil. C'est à travers ce dernier que l'acquisition, le traitement et la communication de données ont lieu. Le nœud-capteur comprend : les unités d'acquisition, de traitement, de communication et d'alimentation [Dargie 2010], voir figure 1.2.

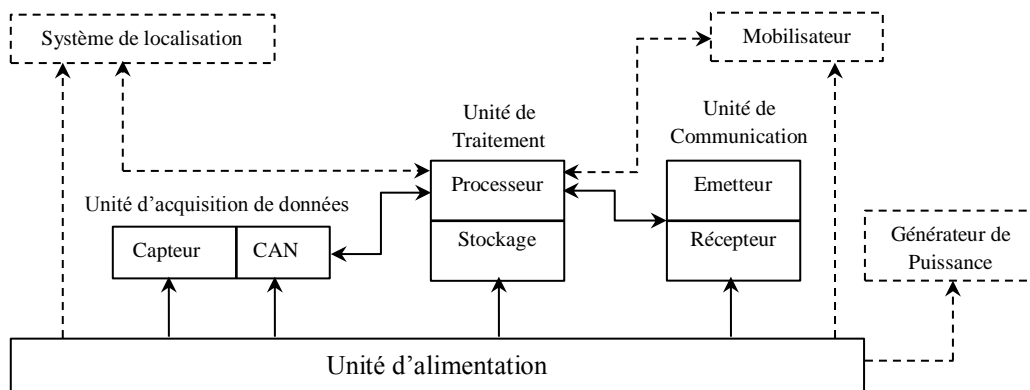


Figure 1.2. Anatomie générale d'un nœud-capteur.

#### 1.2.1.1 Unité d'acquisition

L'unité d'acquisition intègre un ou plusieurs capteurs physiques et fournit un ou plusieurs convertisseurs analogique-numérique (CAN). Elle agit en tant qu'interface entre l'environnement physique et le monde virtuel, c'est-à-dire en collectant des données de l'environnement et en les convertissant de signaux analogiques en signaux numériques.

Avec l'avènement de microsystèmes électromécaniques – *micro electro-mechanical systems* – (MEMS), l'acquisition de phénomènes physiques est devenue un processus omniprésent. De nos jours, il y'a une pléthore de capteurs qui mesurent et quantifient des grandeurs physiques à des prix bon marché.

#### 1.2.1.2 Unité de traitement

L'unité de traitement est l'élément central du nœud-capteur. Elle commande tous les autres unités, notamment l'unité d'acquisition (relevé de mesure) et l'unité de communication (envoi et réception de données). Elle se compose d'un processeur, d'une mémoire non volatile (généralement une mémoire flash interne) pour stocker les instructions de

programme, d'une mémoire vive pour stocker temporairement les données détectées et d'une horloge interne.

### 1.2.1.3 Unité de communication

Cette unité est le responsable de toutes les émissions et réceptions de données via un support de communication radio. Elle facilite les conversations entre les sous-composants du nœud-capteur ainsi que les interactions nœud à nœud.

L'unité de communication est l'unité la plus gourmande en énergie et sa consommation doit être régulée [Akyildiz 2010].

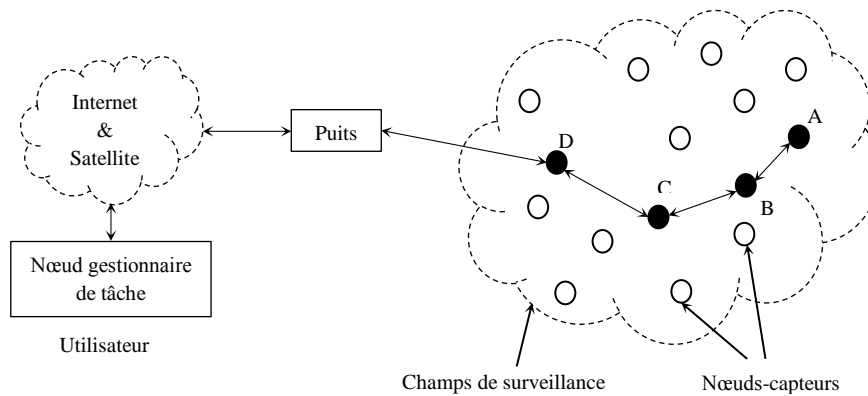
### 1.2.1.4 Unité d'alimentation

Cette unité fournit de l'énergie pour tous les composants du nœud-capteur. Typiquement, il s'agit d'une batterie. Les nœud-capteurs étant de petites tailles, les batteries sont aussi petites et donc les ressources énergétiques sont limitées. Quand la batterie du nœud arrive à épuisement, il faut la remplacer sinon le nœud-capteur cesse de fonctionner. Malheureusement, le remplacement des batteries est difficile voire impossible. La durée de vie du nœud-capteur est donc limitée à la durée de vie de sa batterie.

Les nœuds-capteurs peuvent également être dotés par d'autres modules additionnels comme : un *Système de géolocalisation* afin d'identifier la position géographique (un récepteur GPS par exemple), un *Mobilisateur* qui permet la possibilité aux nœuds de se déplacer à l'intérieur de la zone d'intérêt, et finalement, il est nécessaire qu'un nœud soit maintenu en activité pendant une très longue période de temps, un *Générateur de puissance*, tel que des cellules solaires, serait utile afin de garder le nœud alimenté électriquement sans avoir à changer ses batteries.

## 1.2.2 Architecture de communication

Un réseau de capteurs est généralement constitué d'un ensemble de nœuds-capteurs dispersés dans une zone géographique appelée zone d'intérêt. Chacun de ces nœuds a la capacité de collecter des données et de les router vers une station de base (puits) selon une communication multi-sauts sans infrastructure. Le nœud-puits transmet ces données vers le nœud gestionnaire de tâche via internet ou par satellite. L'utilisateur final peut alors accéder au nœud gestionnaire de tâche pour récupérer les données collectées. Cette architecture [Akyildiz 2002] est montrée sur la figure 1.3.



**Figure 1.3.** Architecture de communication d'un RCSF.

### 1.2.3 Architecture protocolaire

Une pile protocolaire pour les réseaux de capteurs sans fil doit prendre en charge leurs caractéristiques et singularités typiques. Selon [Akyildiz 2002], la pile protocolaire du réseau de capteurs ressemble beaucoup à la pile protocolaire traditionnelle du modèle OSI, avec les couches suivantes: *application*, *transport*, *réseau*, *liaison de données* et *physique*. La couche physique est responsable de la sélection de la fréquence, de la génération de la fréquence porteuse, de la détection du signal, de la modulation et du cryptage des données. La couche liaison de données est responsable du multiplexage des flux de données, de la détection de trame de données, de l'accès au support et du contrôle des erreurs. Elle garantit des connexions fiables point à point et point à multipoint dans un réseau de communication. La couche réseau s'occupe de l'acheminement des données fournies par la couche de transport. La conception de la couche réseau dans les réseaux de capteurs doit prendre en compte l'efficacité énergétique, la communication centrée sur les données, l'agrégation de données, ...etc. La couche de transport aide à maintenir le flux de données et elle peut être importante si un accès aux réseaux de capteurs est prévu via internet ou d'autres réseaux externes. En fonction des tâches d'acquisition, différents types de logiciels d'application peuvent être mis en place et utilisés sur la couche application.

Les réseaux de capteurs doivent également prendre en compte les plans de gestion suivants afin de fonctionner efficacement: plans de gestion de mobilité, d'énergie, de tâches, de qualité de service (QoS) et de la sécurité [Akyildiz 2002, Wang 2010]. La pile protocolaire ainsi que les plans associés utilisés par le puits, les clusterheads et les nœuds-capteurs sont illustrés sur la figure 1.4. Le plan de gestion de l'énergie est responsable de la réduction de la consommation d'énergie et peut désactiver les fonctionnalités afin de préserver l'énergie. Le

plan de gestion de la mobilité détecte et enregistre le mouvement des nœuds de sorte qu'un chemin de données vers le puits soit toujours maintenu. Le plan de gestion des tâches équilibre et planifie les tâches d'acquisition de données affectées au champ de détection. Ainsi, seuls les nœuds nécessaires sont affectés à des tâches d'acquisition, les autres pouvant se concentrer sur le routage et l'agrégation de données. La gestion de la qualité de service dans les réseaux de capteurs [Howitt 2006] peut être très importante s'il existe une exigence en temps réel en ce qui concerne les services de données. La gestion de la qualité de service traite également de la tolérance aux pannes, du contrôle des erreurs et de l'optimisation des performances en termes de certaines métriques de qualité de service. La gestion de la sécurité est le processus de gestion, de surveillance et de contrôle du comportement lié à la sécurité d'un réseau. La fonction principale de la gestion de la sécurité consiste à contrôler les points d'accès aux données critiques ou sensibles. La gestion de la sécurité inclut également l'intégration transparente de différents modules de fonctions de sécurité, notamment le cryptage, l'authentification et la détection d'intrusion.

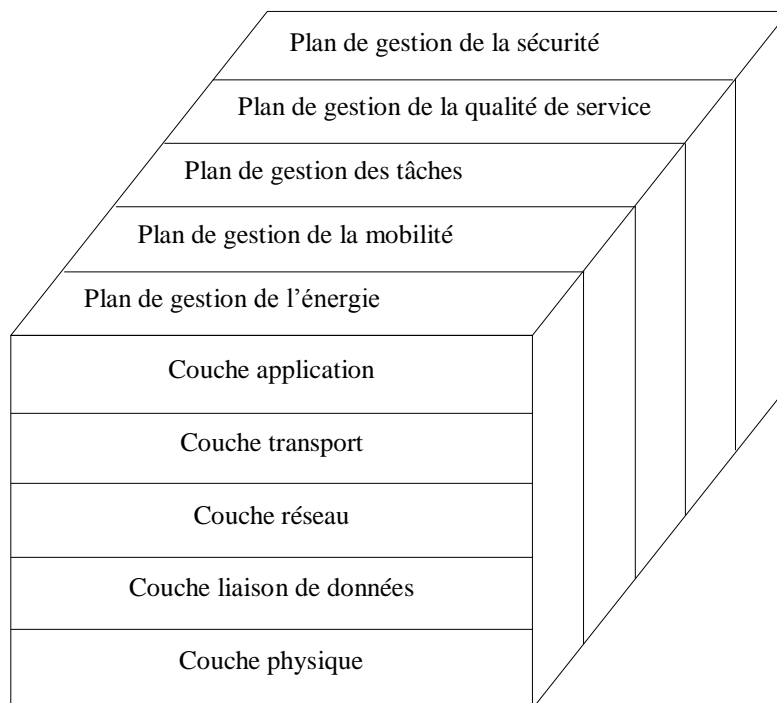


Figure 1.4. Pile protocolaire des réseaux de capteurs sans fil [Wang 2010].

### 1.3 Contraintes de conception des réseaux de capteurs

La conception et le développement des réseaux de capteurs sont influencées par de nombreux facteurs, notamment la tolérance aux pannes; passage à l'échelle; coûts de

production; environnement d'exploitation; topologie de réseau; contraintes matérielles; support de transmission; et consommation d'énergie [[Akyildiz 2002](#), [Akyildiz 2010](#)].

### **1.3.1 Tolérance aux pannes**

La tolérance aux pannes est la capacité de maintenir les fonctionnalités du réseau de capteurs sans aucune interruption due à des défaillances de nœuds-capteurs. Dans un réseau de capteur sans fil, certains nœuds-capteurs peuvent tomber en panne ou être bloqués en raison d'un manque d'alimentation, de dommages physiques ou d'interférences environnementales. La défaillance des nœuds-capteurs ne doit pas affecter la tâche globale du réseau.

### **1.3.2 Passage à l'échelle**

Bien que le déploiement à haute densité des nœuds-capteurs dans les RCSFs fournit une redondance et améliore la tolérance aux pannes du réseau, cela crée également des défis du passage à l'échelle. Le nombre de nœuds-capteurs déployés dans une zone d'intérêt pour étudier un phénomène particulier peut être de l'ordre de centaines ou de milliers. Selon l'application, ce nombre peut également atteindre une valeur extrême de millions. Par conséquent, les protocoles de réseautage développés pour ces réseaux devraient pouvoir manipuler efficacement ce grand nombre de nœuds.

### **1.3.3 Coût de fabrication**

Étant donné que les réseaux de capteurs comprennent un grand nombre de nœuds, le coût d'un seul nœud est très important pour justifier le coût global des réseaux. Si le coût du réseau est plus élevé que le déploiement de capteurs traditionnels, l'utilisation de cette technologie n'est plus justifiée, et elle ne serait pas rentable. En conséquence, le coût de chaque nœud-capteur doit être maintenu bas autant que possible. Le coût d'un nœud-capteur doit être nettement inférieur à 1 \$ pour que le réseau de capteurs soit réalisable [[Rabaey 2000](#)].

### **1.3.4 Environnement d'exploitation**

Les nœuds-capteurs sont déployés de manière très dense dans des zones géographiques éloignées, au fond d'un océan, sur un champ de bataille au-delà des lignes ennemies, attaché

à des animaux, ...etc. Par conséquent, ils doivent pouvoir fonctionner sans surveillance dans des conditions qui peuvent être hostiles à l'homme.

### 1.3.5 Topologie de réseau

La topologie des réseaux de capteurs est dynamique, elle peut être changée au cours du temps à cause de plusieurs raisons : défaillance de certains nœuds en raison d'un endommagement physique ou l'épuisement de l'énergie, changements de la position des nœuds dus à leur mobilité (attaché à un objet mobile par exemple) ou dus à des facteurs externes (vents,...etc.), l'ajout de nouveaux capteurs au réseau implique aussi une remise à jour de la topologie. Ainsi, les nœuds-capteurs doivent être capables d'adapter leur fonctionnement aux différents changements dans le réseau afin de maintenir la topologie souhaitée.

### 1.3.6 Contraintes matérielles

Chaque nœud-capteur est composé de quatre unités de base, comme déjà illustré sur la figure 1.2 : unité de détection, unité de traitement, un émetteur-récepteur radio et une unité d'alimentation. Eventuellement, le nœud-capteur peut également avoir des composants supplémentaires dépendants de l'application tels qu'un système de localisation, un générateur de puissance et un mobilisateur. Cependant, chaque composant supplémentaire vient avec un coût supplémentaire et augmente la taille physique du nœud ainsi que la consommation d'énergie. Ainsi, un composant supplémentaire doit être toujours en balance avec le coût et les exigences de faible puissance.

### 1.3.7 Support de transmission

Dans un réseau de capteurs multi-sauts, les nœuds communicants sont reliés par un support sans fil. Ces liens peuvent être formés par des supports radio, infrarouges ou optiques.

Les liaisons radio consistent à utiliser les bandes ISM (*industrial scientific and medical*), qui offrent une communication sans-licence (*license-free*) dans la plupart des pays. Un autre mode de transmission possible pour la communication inter-nœuds dans les réseaux de capteurs est l'infrarouge, qui a l'avantage d'être sans-licence, robuste aux interférences, et peu onéreux. Le Bluetooth et les communications radio ZigBee sont également utilisées.

Pour permettre le fonctionnement global de ces réseaux dans le monde entier, le support de transmission choisi doit être normé.

### 1.3.8 Consommation d'énergie

Un nœud-capteur sans fil ne peut être équipé que d'une source limitée d'énergie, des piles alcalines par exemple, souvent non rechargeables et leur remplacement s'avère difficile voire impossible. Par conséquent, la durée de vie de réseau, montre une forte dépendance de la durée de vie des batteries de nœuds. Dans un réseau de capteurs multi-sauts, chaque nœud joue deux rôles distincts et complémentaires: collecteur de données et routeur de données. Les opérations liées à chaque rôle affectent la manière dont l'énergie est consommée dans un nœud-capteur. Ainsi, les sources qui consomment de l'énergie pendant le fonctionnement de chaque nœud doivent être analysées et entretenues efficacement.

## 1.4 Classification des réseaux de capteurs

La classification des réseaux de capteurs sans fil dépend de plusieurs critères [Tilak 2002, Karl 2005]. En effet, pour chaque type d'application, ces réseaux peuvent avoir des caractéristiques différentes. Ils se distinguent, généralement, par le modèle de collecte de données, le modèle de communication et le modèle de mobilité dans le réseau.

### 1.4.1 Modèle de collecte de données

Les données du capteur peuvent être collectées selon quatre grands différents scénarios d'applications [Amiya 2010, Cristian 2009] :

- *Application périodique (Time-Driven)* : Dans le mode périodique, les nœuds-capteurs recueillent des informations de l'environnement à des moments prédéterminés et envoient périodiquement les données à la station de base. Les applications de surveillance sont les premières concernées où le but principal est d'avoir une information régulière de la zone surveillée. Dans l'exemple de la figure 1.5-(a), une mesure est capturée périodiquement par le nœud A, puis, les données sont transmises vers le puits à travers les nœuds B, C et D.
- *Application à la demande (On-Demand)* : Dans le mode à la demande, les utilisateurs décident quand collecter les données. Ils envoient des requêtes aux nœuds-capteurs leurs indiquant qu'ils souhaitent recevoir des données, puis attendent que le type de données requis soit envoyé dans le format demandé. Dans l'exemple de la figure 1.5-

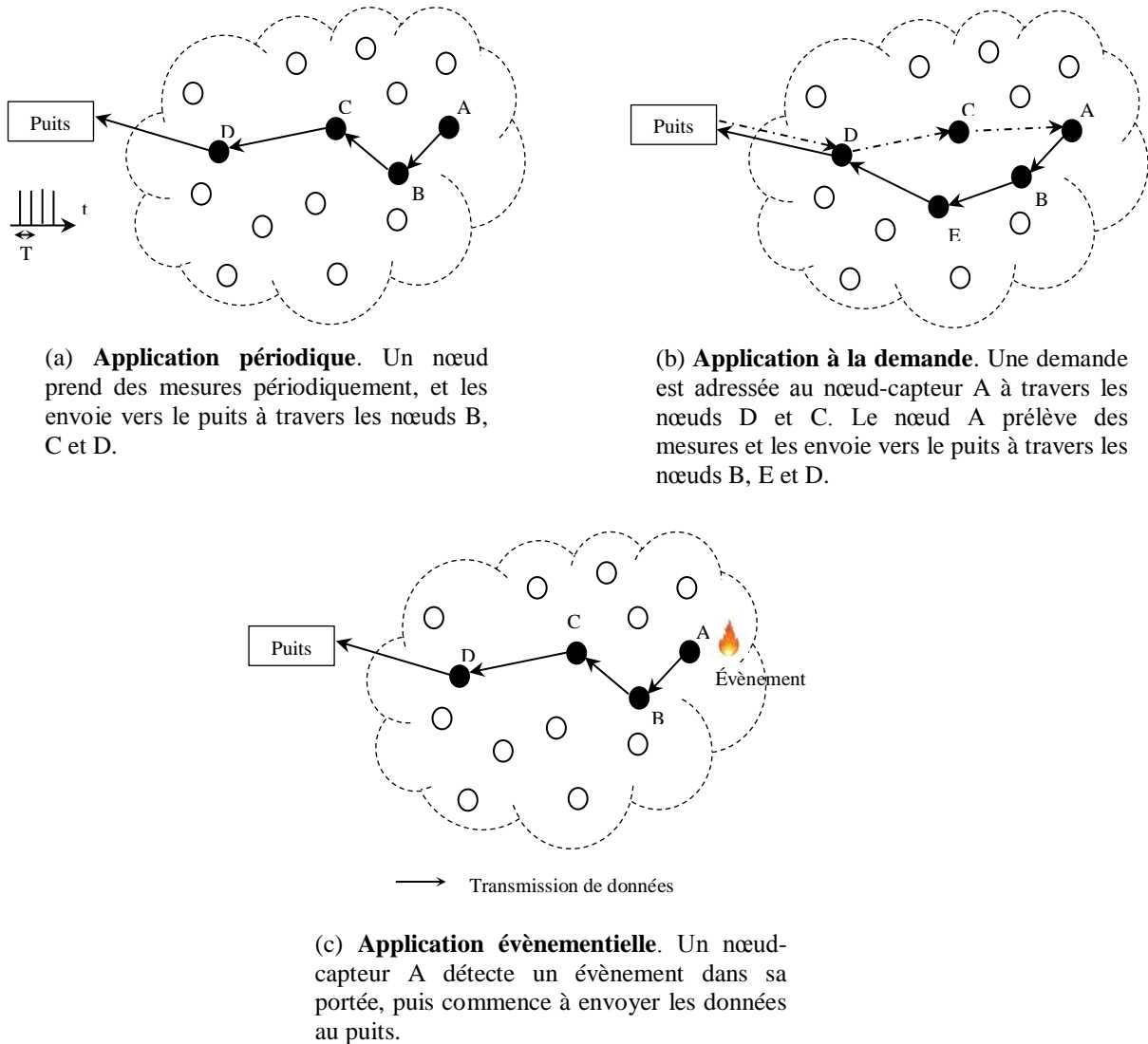
(b), une demande est adressée au nœud source A à travers les nœuds D et C. A la réception, le nœud A, active son unité d'acquisition et envoie ses mesures vers le puits, cette fois par le chemin constitué des nœuds intermédiaires B, E et D.

- Application évènementielles (*Event-Driven*): Dans ce scénario, les nœuds-capteurs signalent les données à la station de base une fois un évènement précis (par exemple, incendie) a été détecté. Dans l'exemple de la figure 1.5-(c), le nœud-capteur A détecte un évènement dans sa portée de détection; puis commence à envoyer les données vers le puits à travers les nœuds B, C et D.
- Applications hybrides : toute alliance des cas précédents.

### 1.4.2 Modèle de communication

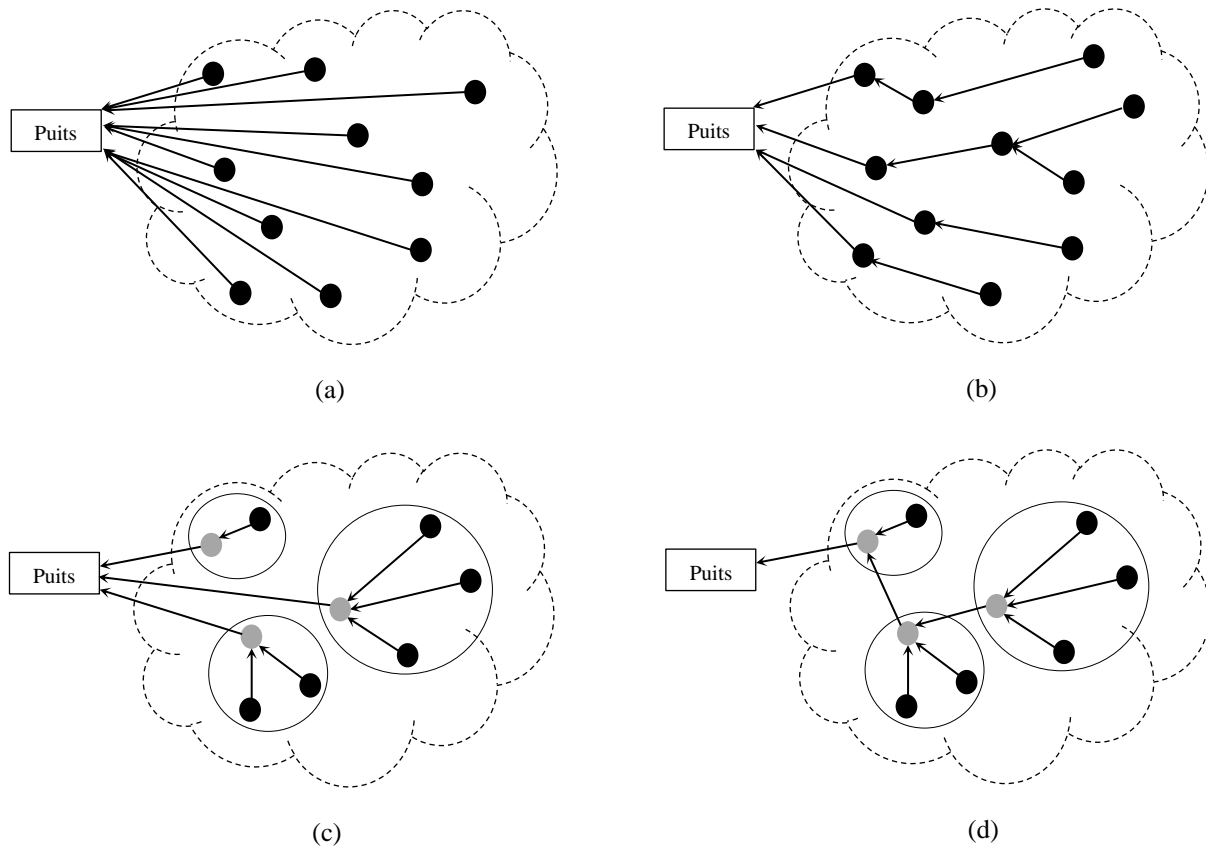
Le modèle de communication adopté dans les réseaux de capteurs dépend de la méthode employée pour faire acheminer les données du capteur vers la station de base. On en distingue trois type: modèle de communication à un seul-saut (single-hop), modèle de communication multi-saut (multi-hop) et modèle de communication hiérarchique [Dargie 2010, Amiya 2010, Benjeddou 2015].

- Dans le modèle de communication à un seul-saut, tous les nœuds-capteurs peuvent communiquer directement avec la station de base. Ce modèle de communication directe est l'approche la plus simple, où toutes les données voyagent en un seul-saut pour atteindre la destination. Dans ce type de réseau les nœuds envoient leurs données en utilisant une forte puissance comme illustré dans la figure 1.6-(a).
- Dans le modèle de communication multi-sauts, les nœuds-capteurs doivent non seulement générer et diffuser leurs propres informations, mais également servir de relais ou de nœuds d'acheminement pour d'autres nœuds-capteurs. Comme le bilan énergétique d'un capteur étant très limité, la portée de transmission des capteurs est également limitée. Ainsi, les réseaux de capteurs fonctionnent généralement de manière multi-saut. La figure 1.6-(b), montre un exemple sur ce type de communication.



**Figure 1.5.** Différents scénarios d'applications pour réseaux de capteurs sans fil.

- Dans le modèle de communication hiérarchique, la zone d'observation est divisée en plusieurs clusters. Dans chaque cluster, il existe un clusterhead. La sélection des clusterheads s'effectue via une candidature globale ou une élection locale. Le clusterhead s'occupe de récupérer les informations auprès des capteurs dans son cluster et de les transmettre directement à la station de base (figure 1.6-(c)) ou par le biais d'une communication multi-saut entre les clusterheads (figure 1.6-(d)).



**Figure 1.6.** Modèles de communication dans un réseau de capteurs.

### 1.4.3 Modèle de mobilité

On peut distinguer deux grandes formes de mobilité :

- *Mobilité des nœuds* : Dans cette forme de mobilité, le réseau de capteur est constitué d'un ensemble de nœuds-capteurs mobile (par exemple, des capteurs attaché à des soldats) et d'un puits fixe. Cette forme convient aux réseaux qui exploitent des zones inaccessibles ou dangereuses. La mobilité des nœuds dans un réseau de capteur entraîne le changement de sa topologie, ce qui entraîne des déconnexions fréquentes. Ainsi, le réseau doit se réorganiser assez fréquemment pour pouvoir fonctionner correctement.
- *Mobilité des puits* : Dans cette forme de mobilité, seulement le point de collecte qui est mobile, les autres nœuds sont fixes. La mobilité des puits a été exploitée pour réduire et équilibrer la dépense énergétique des capteurs [Friedmann 2007]. La mobilité des puits peut être *incontrôlable* ou *contrôlable* [Amiya 2010]. Dans la première, la mobilité est obtenue en attachant un nœud collecteur sur une certaine

entité mobile telle qu'un animal ou un bus navette, qui existe déjà dans l'environnement de déploiement et qui est hors de contrôle du réseau. Dans la deuxième, la mobilité est obtenue en ajoutant intentionnellement une entité mobile (par exemple, un robot mobile) au réseau pour transporter le puits.

## 1.5 Consommation d'énergie d'un nœud-capteur

### 1.5.1 Formes de dissipation d'énergie

Les nœuds-capteurs sont alimentés principalement par des batteries de capacité limitées et non rechargeables. Ils doivent donc fonctionner avec un bilan énergétique frugal. En outre, ils sont le plus souvent déployés dans des zones inaccessibles, il est donc pratiquement impossible de recharger ou de remplacer les batteries des nœuds lorsqu'elles arrivent à épuisement. Ainsi, les différents facteurs provoquant la dissipation de l'énergie pendant le fonctionnement d'un nœud-capteur doivent être analysés. Cette dissipation d'énergie est due essentiellement aux unités suivantes : l'unité d'acquisition de données, l'unité de traitement et celle de communication [Akyildiz 2010, Raghunathan 2002].

#### 1.5.1.1 Unité d'acquisition de données

Il y a plusieurs sources de consommation d'énergie par le module de détection, notamment l'échantillonnage et la conversion des signaux physiques en signaux électriques, le traitement des signaux et la conversion analogique-numérique. Vu la diversité des capteurs, il n'y a pas de valeurs typiques de l'énergie consommée [Kacimi 2009].

#### 1.5.1.2 Unité de traitement

L'unité de traitement consomme généralement peu d'énergie en comparaison avec l'unité de communication. Il est rapporté dans [Pottie 2000], que le coût énergétique nécessaire pour transmettre 1 KB sur une portée de 100 mètres est d'environ 3 joules. En revanche, un processeur à usage général ayant une efficacité énergétique de 100 millions d'instruction par seconde (MIPS) pourrait exécuter  $3 \times 10^8$  instructions pour cette énergie.

L'unité de traitement possède divers modes de fonctionnement : *actif*, *idle* et *sommeil*, employés à des fins de gestion d'énergie. Chaque mode est caractérisé par une consommation d'énergie différente de l'autre mode. Par exemple, le StrongARM consomme 50 mW en

mode *idle*, et juste 0.16 mW dans le mode *sommeil*. Toutefois, la transition entre les modes de fonctionnement implique un surplus d'énergie et de latence [Raghunathan 2002].

Enfin, il faut noter que l'unité de traitement se charge du codage/décodage des paquets de données ainsi que le maintien de tables diverses (tables de voisinage, routage,...etc.) pour les protocoles de communication des couches hautes. La conception des algorithmes et des protocoles de communication va donc influencer largement l'énergie consommée par l'unité de traitement [Makkaoui 2012].

### 1.5.1.3 Unité de communication

La communication est effectuée par le circuit émetteur-récepteur lors de la réception et de la transmission de données. Le nœud-capteur dépense le maximum d'énergie en communiquant avec les autres nœuds au sein d'un réseau.

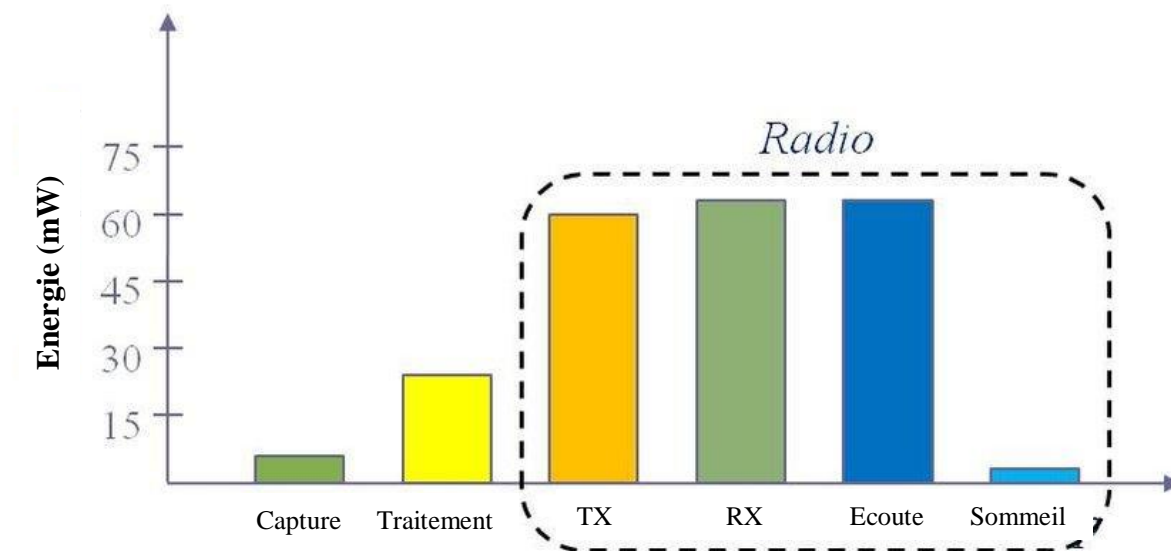
L'émetteur-récepteur radio opère dans quatre modes de fonctionnement : *émission*, *réception*, *écoute (idle)*, et *sommeil (sleep)*. La répartition de la consommation d'énergie de l'émetteur-récepteur radio dans la figure 1.7, montre qu'un émetteur-récepteur dépense une quantité d'énergie presque similaire pour l'émission et la réception, ainsi que lorsqu'il est en mode écoute. Ainsi, une quantité importante d'énergie peut être économisée en mettant l'émetteur-récepteur dans un état de veille (*sleep*), chaque fois que le nœud-capteur n'a pas besoin de transmettre ou de recevoir des données. Cela permet des économies d'énergie allant jusqu'à 99,99% (de 3  $\mu$ W à 59,1 mW) [Akyildiz 2010].

L'histogramme présenté par la figure 1.7, illustre la consommation d'énergie par les différentes unités d'un nœud-capteur [Akyildiz 2010].

### 1.5.2 Sources de surconsommation d'énergie

La surconsommation d'énergie correspond à toute consommation inutile que l'on peut éviter afin de conserver l'énergie d'un nœud-capteur. Les sources de cette surconsommation sont nombreuses et peuvent être due à une ou plusieurs faits suivants. Elles peuvent être engendrées lors de l'acquisition de données lorsque celle-ci est mal gérée, par exemple, une fréquence d'échantillonnage mal contrôlée [Alippi 2007]. Le routage dans les réseaux de capteurs est un routage multi-sauts. L'acheminement des paquets d'une source donnée à une destination se fait à travers plusieurs nœuds intermédiaires. Ainsi, un nœud consomme de l'énergie soit pour transmettre ses données ou pour relayer les données des autres nœuds.

Dans ce contexte, une mauvaise politique de routage peut avoir des conséquences graves sur la durée de vie du réseau. La surconsommation concerne également la partie communication. En effet, cette dernière est sujette à plusieurs phénomènes qui surconsomment de l'énergie surtout au niveau MAC où se déroule le contrôle d'accès au support sans fil [Kacimi 2009]. Certains de ces phénomènes sont les causes majeures de la perte d'énergie et ont été recensés dans [Rezaei 2012, Enzinger 2012, Araar 2015].



**Figure 1.7.** Consommation d'énergie du nœud-capteur MicaZ [Akyildiz 2010].

- *Les collisions* : sont à la fois une source de dégradation des performances du réseau et de perte d'énergie. Les collisions se produisent quand les nœuds reçoivent plusieurs paquets de données en même temps. En conséquence, la donnée reçue est inutile et le processus de transmission doit être répété alors que l'énergie est dissipée. Ces retransmissions peuvent consommer beaucoup d'énergie, car les pertes d'énergie sont multipliées par le nombre de sauts entre la source et la cible. Tous les protocoles MAC essaient à leur manière d'éviter les collisions.
- *La surémission (Overemitting)* : Les envois infructueux se produisent lorsqu'un nœud essaie de communiquer avec un autre nœud qui n'est plus accessible parce qu'il est en mode *sommeil* ou hors de portée par exemple. Le nœud émetteur reste en attente d'un acquittement en provenance du nœud récepteur, si aucune réponse n'a été reçue, celui-ci procède à la retransmission de la même trame plusieurs fois. Les

transmissions répétées augmentent les dépenses en énergie des nœuds individuels pour envoyer leurs données ainsi que le temps de latence de l'ensemble du réseau.

- *L'écoute passive (idle listening)* : L'écoute passive se produit lorsqu'un nœud est à l'écoute du canal alors qu'il y a aucune transmission. Cela peut, de façon abusive, engendrer une consommation considérable et inutile d'énergie surtout dans le cas des réseaux à faible charge de trafic. Plusieurs types de radios présentent un coût en énergie significatif pour le mode « *idle* ». Eteindre la radio est une solution, mais le coût de la transition entre les modes consomme également de l'énergie, la fréquence de cette transition doit alors rester raisonnable.
- *L'écoute abusive (overhearing)* : cette situation se présente quand un nœud reçoit des paquets qui ne lui sont pas destinés. Cela peut engendrer une surconsommation importante d'énergie surtout dans le cas des réseaux très denses et à très fort trafic, particulièrement dans les réseaux « *mostly-on* »<sup>1</sup>.
- *L'overhead des paquets de contrôle* : Plusieurs protocoles de la couche MAC fonctionnent par échange de messages de contrôle (*overhead*) pour assurer différentes fonctionnalités : signalisation, connectivité, établissement de plan d'accès et évitement de collisions. Tous ces messages consomment de l'énergie.

## 1.6 Techniques de conservation de l'énergie

### 1.6.1 Notion de la durée de vie d'un réseau de capteurs

Les études sur la durée de vie sont d'abord venues parce que la recharge ou le remplacement des batteries n'est pas faisable dans de nombreux scénarios (trop de nœuds, environnement hostile, ...etc.), et ainsi, la durée de vie du réseau ne peut être étendue à l'infini. Bien entendu, la durée de vie a été ensuite discutée de différents points de vue, ce qui a conduit au développement de divers paramètres de durée de vie. Selon les sources de consommation d'énergies considérées dans chaque métrique et les besoins spécifiques des applications envisagées. Ces métriques peuvent conduire aux différentes estimations de durée de vie réseau.

---

<sup>1</sup> Réseau dont les radios des nœuds-capteurs sont dans la plupart du temps allumées

La durée de vie du réseau est peut-être la métrique la plus importante pour l'évaluation des réseaux de capteurs. Elle a fait l'objet de plusieurs définitions en fonction des besoins spécifiques des applications. Les définitions possibles et proposées dans la littérature sont données dans [Mahfoudh 2008, Dietrich 2009, Araar 2015] et rappelées ci-dessous.

- *Durée de vie basée sur le nombre de nœuds vivants* : La définition la plus fréquemment trouvée dans la littérature est la durée de vie de  $n$ -sur- $n$ . Dans cette définition, la durée de vie du réseau se termine dès que le premier nœud échoue. Une autre variante de la définition  $n$ -sur- $n$ , définit la durée de vie comme le temps jusqu'à ce que la fraction de nœuds vivants tombe au-dessous d'un seuil prédéfini «  $\beta$  », ou le temps pendant lequel au moins  $k$ -sur- $n$  nœuds sont vivants. Les auteurs dans [Hellman 2006] définissent une autre métrique basée sur le nombre de nœuds disponibles. Ils divisent l'ensemble de nœuds en nœuds critiques et non critiques, puis autorisent  $k$  défaillances de nœuds dans le groupe de nœuds non critiques et aucune défaillance dans le groupe de  $m$  nœuds critique. Ils nomment cette approche, durée de vie de  $m$ -dans- $k$ -sur- $n$ .
- *Durée de vie basée sur le nombre de nœuds morts* : Il est possible de définir la durée de vie réseau comme le temps jusqu'à ce que tous les nœuds épuisent leur énergie [Tian 2002]. Cette métrique est très rarement utilisée. Il s'agit bien entendu du fait que la métrique est beaucoup trop optimiste pour être utile. Dans la plupart des cas, un réseau de capteurs cesse de fournir un service utile pendant une longue période avant que le dernier nœud échoue définitivement.
- *Durée de vie basée sur la couverture* : Compte tenu des caractéristiques spécifiques des réseaux de capteurs, estimer la durée de vie du réseau comme étant le temps pendant lequel la région d'intérêt est couverte par les nœuds-capteurs semble être un moyen naturel pour définir la durée de vie. Plusieurs articles fondent leurs définitions de la durée de vie du réseau sur une variante de couverture. Parmi celles-ci, la définition la plus courante, utilise 1-couverture pour définir la durée de vie comme le temps pendant lequel la région d'intérêt est complètement dans la portée de détection d'au moins un nœud-capteur, c'est-à-dire que la région est couverte par au moins un nœud [Cardei 2005]. Une variante moins stricte de cette définition est que seulement une fraction  $\alpha$  de la région d'intérêt doit être couverte [Wu 2005]. Ceci est communément appelé  $\alpha$ -couverture. Une autre variante plus stricte exigeant que

chaque point dans la zone d'intérêt soit couvert par au moins  $k$  nœuds [Mo 2005]. Ceci est appelé  $k$ -couverture.

- *Durée de vie basée sur la connectivité* : Un autre groupe de métriques prend en compte la connectivité du réseau. La connectivité est une métrique qui est couramment rencontré dans le contexte des réseaux ad hoc, car il n'y a pas de notion de couverture capteur dans les réseaux ad hoc et donc la capacité de transmettre des données vers une destination donnée est plus importante. Il y'a plusieurs variantes de durée de vie reposant sur la connectivité du réseau, parmi lesquelles le pourcentage de nœuds pouvant atteindre la station de base via une communication multi-sauts. Les auteurs dans [Cărbunar 2006] définissent la durée de vie de réseau comme la durée pendant laquelle certains nœuds peuvent maintenir une route vers le puits.

## 1.6.2 Taxonomie des approches de conservation d'énergie

Malgré les progrès technologiques qui ont été faits dans le domaine des réseaux de capteur sans fil, la durée de vie des nœuds-capteurs continue d'être un défi majeur et un facteur clé, exigeant davantage de recherches sur l'efficacité énergétique des plates-formes et des protocoles de communication.

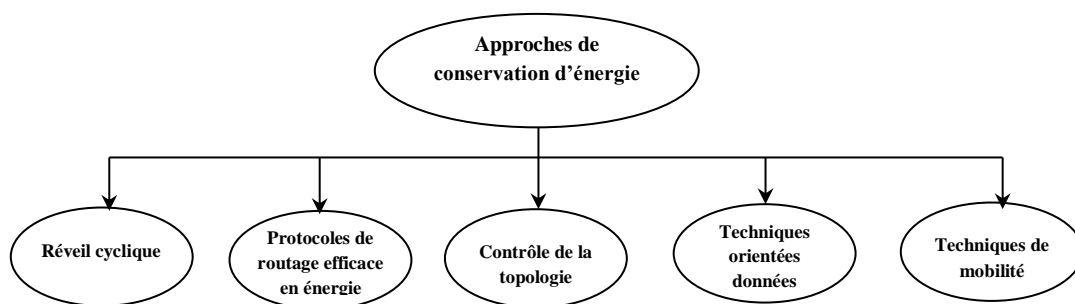
En effet, les concepteurs de solutions pour de tels réseaux se focalisent souvent sur la qualité de service requise par l'application au premier plan pour ensuite assujettir cela avec l'efficacité énergétique. Une telle attention est accordée à l'efficacité énergétique, car le fonctionnement d'un nœud est étroitement lié à la quantité d'énergie disponible dans sa batterie. Ce qui fait que la durée de vie du réseau est en général fortement dépendante des quantités individuelles d'énergie des différents nœuds-capteurs. Ainsi, plusieurs solutions ont été proposées dans la littérature, et utilisent différentes techniques qui peuvent être classées en cinq grandes catégories [Mahfoudh 2008, Anastasi 2009, Rezaei 2012]: approches basées sur le réveil cyclique, protocoles de routage efficace en énergie, contrôle de la topologie, techniques centrées données et finalement, approches basées sur la mobilité, voir figure 1.8.

### 1.6.2.1 Réveil cyclique (*Duty-cycling*)

Le moyen le plus efficace pour conserver de l'énergie est de permettre aux nœuds-capteurs d'aller dormir à chaque fois que la communication n'est pas nécessaire, à condition que les fonctionnalités du réseau et de l'application soient toujours assurées. L'idée est que

les capteurs doivent éteindre leurs radios (c'est-à-dire, les mettre en mode veille à faible consommation d'énergie) en l'absence d'activité réseau et être prêt dès qu'ils ont un nouveau paquet de données à envoyer ou à recevoir, alternant ainsi entre périodes active et sommeil.

Comme les nœuds-capteurs effectuent des tâches en coopération, ils doivent coordonner leurs dates de sommeil et de réveil. Un algorithme d'ordonnancement Sommeil/Réveil accompagne donc tout plan de *Duty-cycling*. Il s'agit généralement d'un algorithme distribué reposant sur les dates auxquelles des nœuds décident de passer entre l'état actif et l'état sommeil [Kacimi 2009].



**Figure 1.8.** Taxonomie des approches de conservation d'énergie dans les RCSFs.

### 1.6.2.2 Protocoles de routage efficaces en énergie

Le routage dans les réseaux de capteurs sans fil est très défiant en raison des caractéristiques intrinsèques qui distinguent ces réseaux à d'autres réseaux sans fil tels que les réseaux mobiles ad hoc ou les réseaux cellulaires. Afin de maximiser la durée de vie du réseau, de nombreuses stratégies se concentrent sur le routage efficace en énergie. Le but de ces stratégies est de trouver une route tenant compte de l'énergie résiduelle des nœuds et de l'énergie consommée par la transmission de paquets de la source à la destination.

Un bon nombre de protocoles de routage proposés dans la littérature, qui minimisent l'énergie dissipée est documenté dans [Al-karaki 2004, Akkaya 2005].

### 1.6.2.3 Contrôle de la topologie

Habituellement, en raison de la densité élevée et le déploiement aléatoire des nœuds, de nombreux capteurs se trouvent très proches les uns des autres. Dans ces cas, les informations collectées que les nœuds très proches fournissent, auront tendance à être très similaire, et par conséquent, la présence de nœuds « redondants » dans le réseau n'est pas tout à fait

nécessaire. En outre, le fait d'avoir des nœuds redondants actifs dans le réseau n'est même pas désiré; ils dissipent d'énergie supplémentaire en retransmettant les paquets qui entrent en collision, en raison de la hausse du trafic réseau. Une des solutions à tous ces problèmes consiste à modifier la topologie du réseau en tirant parti de l'adaptabilité de la puissance de transmission des capteurs et de leurs différents modes de fonctionnement. La combinaison correcte de la puissance de transmission et de l'état des capteurs permet à l'opérateur de réseau de concevoir une topologie à faible consommation d'énergie tout en préservant la connectivité et la couverture souhaitée de détection et de communication.

Le contrôle de la topologie est ainsi la réorganisation et la gestion des paramètres de nœud et des modes de fonctionnement de temps à autre afin de modifier la topologie du réseau dans le but de prolonger sa durée de vie tout en préservant les caractéristiques importantes, telles que la connectivité et la couverture réseau.

Une classification de différents mécanismes de contrôle de la topologie est documentée dans [Labrador 2009].

#### **1.6.2.4 Techniques orientées données**

La plupart des techniques de conservation d'énergie employées dans les réseaux de capteurs sans fil ne tiennent pas compte des données prélevées par les nœuds. Par conséquent, des approches orientées données peuvent être utiles pour améliorer l'efficacité énergétique encore plus. En fait, l'acquisition (ou prélèvement) de données influe sur la consommation d'énergie des nœuds-capteurs de deux manières:

- *Echantillonnage inutile* : comme les échantillons mesurés peuvent être corrélés, la quantité de données à acquérir peut être réduite davantage en exploitant ces corrélations spatiales et/ou temporelles. Il n'est donc pas nécessaire de communiquer les informations redondantes à la station de base faisant diminuer ainsi la consommation d'énergie du sous-système d'acquisition.
- *Consommation électrique du module d'acquisition* : elle peut non seulement être pertinente, mais elle peut aussi être supérieure à la consommation d'énergie de la radio ou même supérieure à la consommation d'énergie du reste du nœud-capteur [Alippi 2007]. Dans ce cas, réduire la communication ne suffit pas quand le capteur lui-même est gourmand en énergie.

Les techniques orientées données regroupent les techniques de réduction de données qui traitent le cas des échantillons inutiles et les techniques basés sur l'acquisition des données efficace en énergie visant principalement à réduire l'énergie dépensée par le sous-système d'acquisition.

### 1.6.2.5 Techniques de mobilité

La plupart des travaux sur les réseaux de capteurs sans fil, en particulier, ceux dans les premiers stades de recherche, ont assumés une architecture de réseau de capteurs où les nœuds sont supposés être statiques et leur densité devrait être suffisamment grande pour permettre la communication entre deux nœuds quelconques, éventuellement en utilisant un chemin multi-sauts. Cependant, plus récemment, la mobilité a été considérée comme une solution alternative pour la collecte de données efficace en énergie dans les réseaux de capteurs sans fil. L'objectif de la mobilité est de pouvoir garantir un équilibrage des charges en ce qui concerne la consommation énergétique, et aussi alléger la tâche de routage.

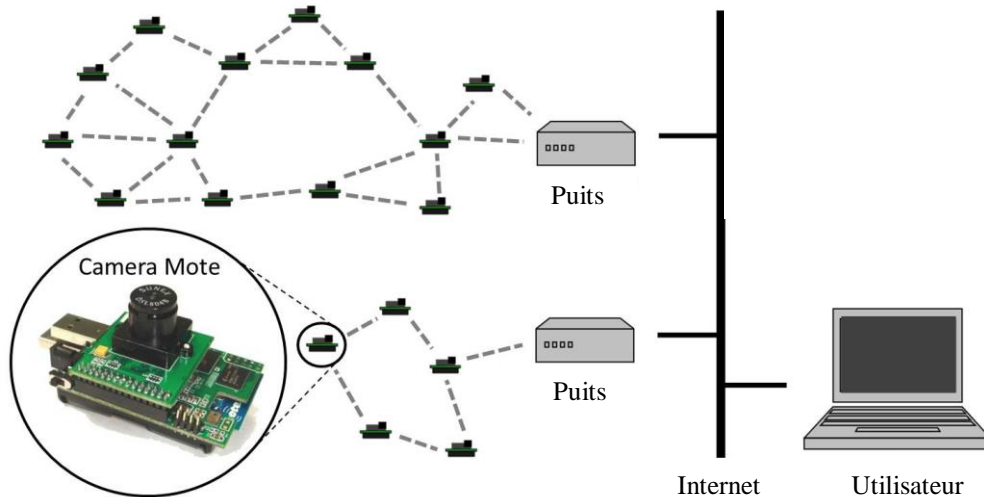
Les techniques de mobilité peuvent être classées en fonction de la nature de l'élément mobile en deux catégories : techniques basées sur la mobilité de la station de base et techniques reposant sur la mobilité des nœuds-capteurs.

## 1.7 Réseau de capteurs d'images

Les progrès récents dans les domaines des systèmes micro-électromécaniques, de technologie de communication sans fil, et de caméras en technologie CMOS (*complementary metal oxide semiconductor*) à faible coût et à faible consommation d'énergie, tels la caméra Cyclops [Rahimi 2005], ont conduit à l'émergence d'un nouveau paradigme, appelé réseaux de capteurs d'images [Akyildiz 2007, Soro 2009]. Un réseau de capteurs d'images est un réseau ad hoc formé d'un ensemble de nœuds-caméras autonomes pouvant communiquer entre eux sans liaison filaire. Chaque nœud-capteur d'images a la capacité d'acquérir, de compresser et de transmettre des images utiles à la station de base (puits), à travers les nœuds voisins auxquels il est connecté, voir figure 1.9.

Un réseau de capteurs d'images n'est qu'un cas spécifique de réseaux de capteurs sans fil ; et une grande partie de la théorie et de l'application des réseaux de capteurs sans fil s'applique aussi aux réseaux de capteurs d'images. Contrairement aux réseaux de capteurs traditionnels, les réseaux de capteurs d'images présentent des défis uniques principalement en

raison de la grande quantité de données à capturer, à traiter et à transmettre au-dessus d'un réseau à ressources restreintes.



**Figure 1.9.** Exemple d'un réseau de capteurs d'images.

### 1.7.1 Spécificités des réseaux de capteurs d'images

Les réseaux de capteurs d'images diffèrent de leurs prédécesseurs, les réseaux de capteurs scalaires principalement dans ce qui suit [Makkaoui 2012, Mammeri 2012, ZainEldin 2015, Araar 2015] :

- La quantité d'information associée à une image est plus grande que celle réservée à une information scalaire simple. En effet, les données fournies par un capteur scalaire sont fournies généralement sur quelques bits (par exemple, la carte d'acquisition MTS400 de Crossbow code les valeurs de température et d'humidité sur 14 bits et les valeurs de luminosité sur 12 bits). Dans ce cas, le transport de ces données nécessite seulement un seul paquet et la compression n'est pas donc exigée. Au contraire, une image conduit à l'emploi de plusieurs centaines ou milliers d'octets et donc son transport nécessite typiquement plusieurs paquets.
- La perte de paquets est tolérée dans les réseaux de capteurs d'images à cause de la redondance de flux visuel, tandis que la perte de données dans les réseaux de capteurs scalaire peut affecter la précision des données collectées. En effet, les images naturelles ont des corrélations spatiales assez marquées entre les pixels voisins. Cela veut dire qu'il est possible de reconstruire une version approximative de l'image

originale même si une partie des informations est perdue dans le réseau. Ainsi, le protocole de communication de bout en bout n'a pas besoin de fournir un service de transport totalement fiable, et des économies d'énergie peuvent être atteintes à ce niveau.

- Le champ de perception des capteurs scalaires est omnidirectionnel, c'est-à-dire que l'orientation du capteur n'influe pas sur la valeur de la grandeur physique mesurée. Le champ de vision des nœuds-capteurs d'images est par contre restreint en direction.
- Contrairement aux réseaux de capteurs scalaires, les réseaux de capteurs d'images font appel à des ressources importantes en mémoire, en traitement et en puissance de communication pour pouvoir manipuler la grande quantité du flux visuel.
- Les nœuds-capteurs d'images voisins surveillant la même petite région locale d'intérêt ont de multiples et de différentes vues de cette scène, comparativement aux capteurs de données scalaires où une valeur unique (par exemple, température) est collectée par tous les nœuds voisins (situés dans la même région).

## 1.7.2 Plateformes de capture d'image

### 1.7.2.1 Caméras basées sur des composants commerciaux

Plusieurs nœuds et cartes de capture d'images dédiés pour les réseaux de capteurs sans fil ont été développés à base de composants commerciaux dits COTS (*Commercial Off-The-Shelf*), par exemple, des imageurs CMOS, des microcontrôleurs, des mémoires, ...etc. Quelques plateformes sont détaillées dans ce qui suit.

#### – Cyclops

La caméra Cyclops [Rahimi 2005] a été développée par les laboratoires Agilent et le CENS (*Center for Embedded Network Sensing*) de l'UCLA (*Université de Californie à Los Angeles*). La philosophie derrière l'approche utilisée dans la caméra Cyclops consiste à séparer les ressources utilisées pour l'imagerie de celles affectées à la communication. Ainsi, Cyclops consiste en un processeur RISC 8-bits (ATmega128L d'ATMEL), un CPLD, une SRAM externe (64 Ko) et une mémoire FLASH externe (512 Ko). Le module de caméra est un imageur CMOS (ADCM-1700 d'Agilent) ultra-compact de résolution CIF (352×288 pixels). La caméra Cyclops possède un connecteur de 51 pins lui permettant d'être enfichée

sur un mote Mica2 ou MicaZ. La figure 1.10 montre une caméra Cyclops attachée à un nœud Mica2.



**Figure 1.10.** Caméra Cyclops attachée à un mote Mica2.

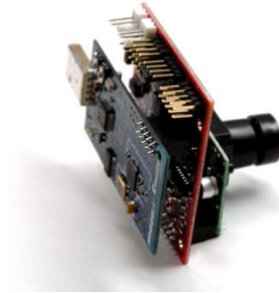
– *MeshEye*

Ce nœud-caméra intelligent sans fil [Hengstler 2007] est une extension du travail présenté dans [Downes 2006]. Il est principalement destiné aux applications du domaine de la surveillance intelligente distribuée. Sa conception se caractérise par la mise en place d'une interface directe entre le processeur intégré et plusieurs capteurs d'image. Le cœur du système est un processeur ARM7 32-bits (AT91SAM7S d'ATMEL). Jusqu'à huit imageurs de 1 Kilo-pixels peuvent être hébergés sur le nœud. Un module de caméra VGA (640×480 pixels) est également pris en charge. La connexion sans fil avec les autres nœuds est établie via un émetteur-récepteur radio conforme à la norme 802.15.4 (CC2420 de Texas Instrument). Enfin, une carte mémoire flash (MMC/SD) fournit suffisamment de mémoire non volatile pour la mise en mémoire tampon temporaire des images et même pour l'archivage d'images.

– *CMUCam3*

La CMUcam3 est une plateforme de vision embarquée programmable [Rowe 2007]. Elle a été développée à l'Institut de robotique de l'Université Carnegie Mellon et c'est la dernière d'une série de caméras embarquées. Elle est composée de trois modules principaux: un capteur d'images Omni-vision OV6620 ou OV7620 en technologie CMOS, une mémoire tampon AL4V8M440 de 1 Mo, cadencée à 50 MHz et un microcontrôleur NXP LPC2106. Le LPC2106 est un ARM7TDMI 32-bits opérant à 60 MHz doté de 64 Ko de RAM et de 128 Ko de mémoire flash. La CMUcam3 supporte la résolution CIF avec un capteur de couleur RGB et peut faire un peu de traitement d'image de base sur son propre processeur. La figure 1.11 montre une caméra CMUcam3 attachée à un nœud Tmote Sky.

Des bibliothèques Open Source et des exemples de programmes sont fournis pour développer des programmes en langage C pour la caméra. Il est possible de la connecter à des nœuds-capteurs sans fil tels que Tmote Sky et FireFly.



**Figure 1.11.** CMUCam3 attachée à un mote Tmote Sky.

– *Plateforme basée sur iMote2*

Cette plate-forme de vision pour réseaux de capteurs sans fil est assez simple [Teixeira 2006]. Elle consiste d'une interconnexion directe via une interface série entre le nœud iMote2 et un capteur d'images CMOS OV7649 omni-vision. Cet imageur peut capturer des images couleur VGA (640×480 pixels) à 30 fps et QVGA (320×240 pixels) à 60 fps avec une consommation d'énergie de 44 mW. En mode entièrement actif, avec un mote iMote2 fonctionnant à 104 MHz et un taux d'image de 8 fps, le système consomme 322mW. Une image de cette plateforme est illustrée sur la figure 1.12.



**Figure 1.12.** Plateforme basée sur iMote2.

– *CITRIC*

La configuration générale de cette plate-forme [Chen 2008] est également très similaire à celle de Cyclops, voir figure 1.13. Il y'a un processeur (PXA270 de Marvell) dédié au traitement visuel qui reçoit les images d'un capteur d'images CMOS OV9655 omni-vision de 1280×1024 pixels. Ce processeur communique directement avec un mote TelosB afin de

doter le système de communication sans fil. L'intégration d'un microphone est une caractéristique intéressante de CITRIC. Les informations captées par ce microphone sont également analysées par le processeur de vision, qui devient alors un processeur multimédia. Cela élargit considérablement le champ d'application de cette plateforme.



**Figure 1.13.** Caméra CITRIC avec un mote TelosB.

### 1.7.2.2 Caméras conçues spécifiquement pour les RCSFs

Une proposition différente pour la conception de capteur d'image numérique a été introduite par [Culurciello 2006]. Le capteur d'image ALOHA, un imageur en technologie CMOS conçu spécifiquement pour les réseaux de capteurs sans fil utilise une représentation numérique de l'information basée sur des événements. L'idée de base de cette technique est simple : des événements sont exécutés lorsque des pixels individuels atteignent un seuil de tension déterminé (proportionnel à la quantité de lumière collectée par la surface photosensible). Le pixel envoie une requête à un circuit spécifique pour lui permettre de transmettre sa valeur. En cas d'accès multiples au bus de données, la technique de résolution de problème d'accès ALOHA est employée (chaque pixel envoie ses données sans se préoccuper des autres. S'il y a collision, tous les pixels intervenants doivent retransmettre leurs valeurs ultérieurement). La figure 1.14 montre un imageur ALOHA enfiché sur un nœud Mica2.

Jusqu'à présent, deux versions du capteur d'images ALOHA ont été développés :

- ALOHAim1, un capteur d'image disposant d'une matrice de  $32 \times 32$  pixels et d'un circuit détecteur de contention (ou collision) analogique [Culurciello 2004].
- ALOHAim2, dispose d'une matrice de  $64 \times 64$  pixels (4 quadrants de  $32 \times 32$  pixels) et d'un détecteur de collision numérique [Culurciello 2006].



**Figure 1.14.** Un imageur ALOHA attachée à un mote Mica2.

Les auteurs dans [Gasparini 2011] ont présenté un nœud conçu spécifiquement pour les réseaux de capteurs sans fil. L'objectif de cette approche était d'atteindre des économies d'énergie significatives à tous les niveaux (capture et traitement de l'image, transmission/réception, ...etc.). Le nœud comporte cinq composants logique principaux : une unité de contrôle, qui gère l'ensemble du nœud ; un imageur CMOS ; un dispositif de mémoire pour la mise en mémoire tampon des images ; une unité de traitement qui traite les images acquise ; un émetteur/récepteur radio pour la communication sans fil. Le prototype a été implémenté en utilisant une carte FPGA (unité de contrôle, unité de traitement et la mémoire) et un capteur d'images CMOS de faible consommation d'énergie de  $128 \times 64$  pixels.

### 1.7.3 Exemples d'application

Les réseaux de capteurs d'images sans fil ont une large gamme d'applications qui comprennent la gestion des catastrophes, la surveillance de la faune et de la flore, les applications militaires (détection d'intrusions, localisation et traque de cible), la surveillance de l'environnement (contrôle des processus industriels, machines), la surveillance de l'habitat (enfants, personnes âgées), la réglementation et gestion du trafic routier, ...etc. Certaines de ces applications sont données ci-dessous :

#### 1.7.3.1 Surveillance de la faune et de la flore

Les réseaux de capteurs d'images peuvent être très utiles pour la surveillance de la faune et de la flore dans la nature. Les nœuds peuvent être déployés dans des endroits stratégiques (par exemple, près des nids, des abreuvoirs et des réserves d'eau naturelles) pendant de longues périodes (de l'ordre de plusieurs mois), afin d'obtenir des images utiles sur certaines espèces animales ou végétales sans avoir besoin de se déplacer physiquement.

Quelques expériences peuvent être signalées pour ce type d'application pour l'observation de la faune et de la flore en milieu naturel. Par exemple, le projet *PODS* [Biagioni 2002] développé par l'université d'Hawaii pour l'étude à long terme des espèces végétales rares et en danger. Les capteurs d'images nommés *PODS* sont déployés dans le parc national des volcans d'Hawaii.

Le projet *BearCam* [Wawerla 2008] est un autre exemple d'application pour l'observation de la faune et de la flore. Ce projet qui a été déployé en automne 2005 près de cercle polaire arctique, avait comme objectif de surveiller les ours bruns (grizzly) juste à la sortie de leur sommeil hivernal, voir figure 1.15. Cette application aide les biologistes à collecter des données utiles pour leurs études sur le comportement des ours. De même, comme rapporté dans [Peak 2014], un réseau de capteurs d'images composé de 19 caméras *Cyclops* a été déployé dans la réserve des montagnes James San Jacinto, Californie, dans des nids d'oiseaux. L'objectif était de surveiller le comportement de certains oiseaux pendant la saison de nidification. Les biologistes ont pu trouver en ligne et en temps quasi-réel, l'accès aux images qui peuvent être utiles pour obtenir des données sur la réponse à leurs questions biologiques relatives au dépôt, incubation et éclosion des œufs.



**Figure 1.15.** Projet *BearCam* pour l'observation des ours bruns « grizzly ».

### 1.7.3.2 Evitement, réglementation et contrôle de la circulation

Il est possible de surveiller le trafic routier dans les grandes villes ou les autoroutes et de déployer des services offrant des recommandations afin d'éviter les encombrements. Les capteurs multimédias peuvent également surveiller le flux de circulation des véhicules sur les autoroutes et récupérer des informations globales telles que la vitesse moyenne et le nombre

de voitures. Les capteurs d'images pourraient mettre en mémoire tampon les images en cas d'accident pour une analyse ultérieure de la scène de l'accident.

Dans [Campbell 2005], les auteurs ont présenté un système intelligent d'assistance au stationnement basé sur les réseaux de capteurs d'images qui permet de fournir un service de localisation des places de stationnement disponibles à proximité d'une destination souhaitée, améliorant ainsi la mobilité dans les zones urbaines. Un ensemble de caméras détectent la présence de voitures dans les espaces et mettent à jour une base de données distribuée, de sorte qu'un système de navigation permettant de trouver les espaces disponibles.

### 1.7.3.3 Application militaires

Les réseaux de capteurs d'images peuvent être utilisés pour avertir au préalable les unités de commandement et de contrôle militaires de la survenue d'événements d'intérêt dans des régions hostiles. Dans l'expérience menée à la base MacDill de l'US air force à Tampa, Floride, « *Une ligne dans le sable* » [Arora 2004], un réseau de capteurs d'images a été déployé par une équipe de recherche de l'université d'Ohio. L'objectif du réseau était de détecter et suivre les mouvements d'objets mobiles intrus. Trois différents groupes d'objectifs ont été classés en tenant compte des caractéristiques détectables telles que leur quantité de métal et de leur rapidité de mouvement : personne non armée, soldat et véhicule blindé. Le réseau était constitué de plus de 90 nœuds dotés chacun d'un capteur de métaux, d'un radar doppler et d'une caméra.

### 1.7.3.4 Télé-relevé des compteurs de mesure

Les réseaux de capteurs d'images sans fil peuvent être très utiles pour les applications de mesures dans le domaine de la télérelève des compteurs d'eau, de gaz ou de l'énergie. Un réseau de capteurs d'images a été utilisé dans l'étude de cas: l'utilisation de capteurs d'images sans fil au télé-relevé des compteurs d'eau [Ferrigno 2005]. Le capteur a été convenablement conçu pour être installé pour couvrir la face avant de compteurs traditionnels et d'être à distance interrogé par un opérateur humain pour télécharger une photo du tableau, sans accéder au site d'installation, voir figure 1.16. Le capteur est supposé être interrogé deux fois par mois et les piles doivent être changées avec une fréquence aussi basse que possible d'une fois chaque année.



**Figure 1.16.** Image d'une lecture d'un compteur d'eau.

### 1.7.3.5 Assistance automatisée aux personnes âgées

Les réseaux de capteurs d'images peuvent être utilisés pour la prise en charge et le soutien des personnes âgées. L'utilisation de la combinaison audio et image, qui sont les principales sources d'informations pour la perception humaine, est présentée dans [Silva 2008]. Les auteurs présentent les possibilités de la détection intelligente en utilisant une multitude de capteurs tels que des capteurs audio et visuels afin de détecter les mouvements humains. Ceci peut être appliqué aux soins à domicile et à la sécurité à domicile dans un environnement intelligent.

## 1.8 Conclusion

Dans ce chapitre, nous avons présentés un aperçu de l'état de l'art sur les réseaux de capteurs sans fil, identifiés comme l'une des technologies les plus prometteuses de ce siècle. Concrètement, nous avons évoqué l'architecture matérielle, protocolaire et de communication ; les contraintes de conception et les modèles de fonctionnement de cette technologie importante.

A travers ce chapitre, différents facteurs provoquant la dissipation d'énergie ainsi que les causes majeures de la surconsommation d'énergie d'un nœud-capteur ont été recensées. Quelques approches de conservation d'énergie dans les réseaux de capteurs sans fil trouvés dans la littérature ont également présentées.

Nous avons présentés un cas spécifique des réseaux de capteurs sans fil : les réseaux de capteurs d'images, qui en comparaison avec les réseaux de capteurs classiques qui manipulent des données scalaires simples comme la température ou l'humidité, ces réseaux

présentent des défis uniques principalement en raison de la grande quantité de données à capturer, à traiter et à transmettre au-dessus d'un réseau soumis à des contraintes très strictes en terme de ressources.

Dans le prochain chapitre, nous allons donner une vue d'ensemble des différents techniques de compression d'images proposées dans la littérature. Nous discutons brièvement leurs avantages et inconvénients, ainsi que leurs éventuelles applications dans le domaine des réseaux de capteurs.



# Chapitre 2

## Chapitre 2 : Compression d'images à la source dans les réseaux de capteurs sans fil

### Sommaire

---

<b>2.1</b>	<b>Introduction .....</b>	<b>35</b>
<b>2.2</b>	<b>Classification des techniques de compression d'images .....</b>	<b>35</b>
<b>2.3</b>	<b>Algorithmes de compression sans perte.....</b>	<b>36</b>
2.3.1	Dé-corrélation .....	36
2.3.2	Codage entropique.....	37
<b>2.4</b>	<b>Algorithmes de compression avec pertes.....</b>	<b>38</b>
2.4.1	Algorithmes non-basés sur une transformée .....	38
2.4.2	Algorithmes à base de transformée.....	39
<b>2.5</b>	<b>Conclusion.....</b>	<b>52</b>

---

## 2.1 Introduction

La plupart des études sur les réseaux manipulant des données scalaires ont typiquement assumées que les coûts de calcul, comprenant l'acquisition, prétraitement et la compression, sont insignifiant comparés aux coûts de la communication [Krishnamachari 2002]. Cette hypothèse peut convenir aux capteurs de données scalaires manipulant des paramètres physiques légers, telle que l'humidité, pour lesquels le coût de la compression des données, si elle est effectuée, est négligeable par rapport au coût de la communication. Dans le cas de réseaux de capteurs d'images, cette hypothèse n'est plus remplie, car la grande quantité de flux visuels nécessite toujours de la compression [Mammeri 2012].

Habituellement, la transmission d'images précédée par compression est un choix idéal afin de gagner en temps et en énergie. Bien que des algorithmes de compression standard aient fourni d'excellentes performances en termes de rapport débit-distorsion, ils ne sont malheureusement pas éligibles dans le contexte des réseaux de capteurs. Ces algorithmes sont, en général, utilisés à des fins de stockage, ou encore, utilisés quand aucune restriction du temps ou d'énergie ne soit imposée [Mammeri 2012]. Par exemple, comme rapporté dans [Ferrigno 2005], les performances de plusieurs algorithmes de compression d'images bien connus comme JPEG, JPEG2000 et SPIHT ont été évalué sur un nœud-capteur. Les résultats ont montré que ces algorithmes consomment plus d'énergie en compressant l'image que de la transmettre sans compression. Dans de tel cas, la compression n'est pas justifiée car la transmission de l'image non compressée consomme moins d'énergie.

Nous remarquons dans ce domaine, les efforts de plusieurs auteurs pour adapter les algorithmes de compression d'images standards, comme JPEG ou JPEG2000, aux contraintes inhérentes aux réseaux de capteurs sans fil.

## 2.2 Classification des techniques de compression d'images

Différentes classifications des techniques de compression d'images sont trouvées dans la littérature [Mammeri 2012, ZainEldin 2015, Suseela 2018]. En général, ils sont classés en termes de perte ou non-perte de données; ou en fonction de compression locale et compression distribuée. Notre objectif n'est pas d'examiner l'ensemble de toutes ces approches, mais plutôt de s'intéresser aux algorithmes de la compression locale pour lesquels leurs applications dans le domaine des réseaux de capteurs semblent pratiques.

Les algorithmes de compression d'images peuvent être classés en techniques de compression avec et sans perte, comme présenté sur la figure 2.1. Si la suppression de la redondance est un processus réversible et n'entraîne aucune perte d'informations, on parle alors de compression sans perte. D'autre part, s'il y a une perte d'information, on l'appellera technique de compression avec pertes. Les techniques de compression avec pertes conviennent aux applications où la perte de données redondantes est tolérable et permettent d'atteindre des taux de compression très élevés tout en préservant une qualité visuelle acceptable de l'image reconstruite. En revanche, la compression sans perte convient aux applications où la dégradation de l'image n'est pas tolérable (par exemple, le domaine de l'imagerie médicale) et permet d'atteindre des taux de compression moins élevés que celle avec pertes.

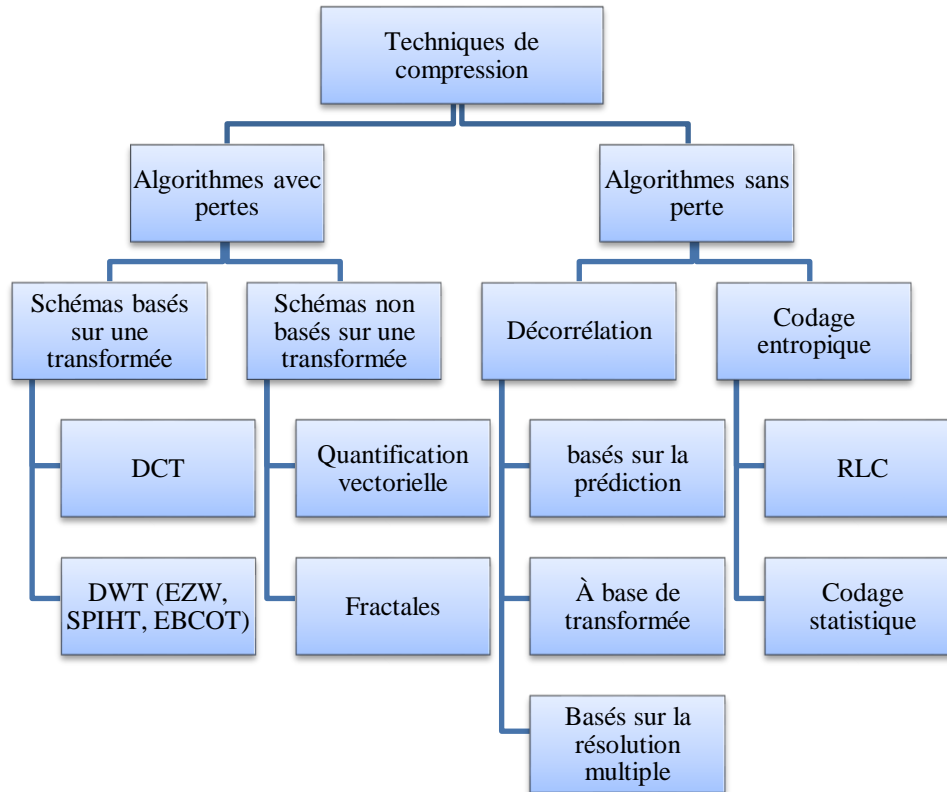
L'avantage de la compression avec pertes par rapport à celle sans perte est de réduire le temps de codage/décodage, d'augmenter le taux de compression et de minimiser la consommation d'énergie en cas d'applications limitées par l'énergie. C'est pourquoi, nous pensons que les algorithmes de compression avec pertes sont fortement encouragés dans le contexte des réseaux de capteurs par rapport aux techniques sans perte. Toutefois, si les algorithmes de compression sans perte donnent les mêmes résultats que ceux avec pertes en termes de dissipation d'énergie, les algorithmes sans perte sont alors encouragés [Mammeri 2012].

## 2.3 Algorithmes de compression sans perte

La compression d'images sans perte peut toujours être modélisée comme une procédure en deux étapes [Yang 2005] :

### 2.3.1 Dé-corrélation

La corrélation entre les échantillons, qui est présentée dans presque tous les types de signaux, représente des informations redondantes qui n'ont pas besoin d'être transmises si des techniques de dé-corrélation réversibles sont appliquées. La dé-corrélation, également connue sous le nom de «blanchiment», peut être accomplie par de nombreuses techniques. Ces techniques sont classées en trois catégories: techniques basées sur la prédiction [Shukla 2010], techniques basées sur la transformation et techniques basées sur la résolution multiple [Carreto-Castro 1993].



**Figure 2.1.** Classification des algorithmes de compression d'images [ZainEldin 2015].

### 2.3.2 Codage entropique

Une fois les données ont été dé-corrélées, plus de compression peut être réalisée en appliquant le codage entropique. Un codage entropique ou compression réversible d'une source de données correspond à un codage sans perte des symboles dont le but est d'éliminer les redondances d'informations présentes dans la source. Le concept fondamental du codage entropique consiste à attribuer les mots de code les plus courts aux symboles qui apparaissent le plus souvent et les plus longs à ceux qui apparaissent rarement. La plupart des codeurs entropique emploie le codage RLC (*Run Length Coding*), le codage statistique (tels que le codage de Huffman ou le codage arithmétique) ou encore le codeur Lempel-Ziv-Welch (LZW).

Comme mentionné précédemment, les techniques de compression sans perte n'entraînent aucune perte de données ou d'informations sur l'image. Ainsi, la taille de l'image compressée reste importante, ce qui rend sa transmission sur le réseau de capteurs très consommateur d'énergie et de bande passante [Mammeri 2012, ZainEldin 2015]. Par conséquent, la compression sans perte n'est pas préférée pour la transmission d'images sur

les réseaux de capteurs, ce qui nous amène à nous concentrer sur les techniques de compression avec pertes qui sont fortement encouragées pour le cas des réseaux de capteurs.

## 2.4 Algorithmes de compression avec pertes

Les techniques de compression avec pertes fournissent un taux de compression élevé par rapport à celles sans perte. Dans la compression avec pertes, l'image compressée n'est généralement pas la même que celle d'origine, mais constitue une approximation proche de l'image originale. Par conséquent, une certaine forme de mesure de distorsion est nécessaire.

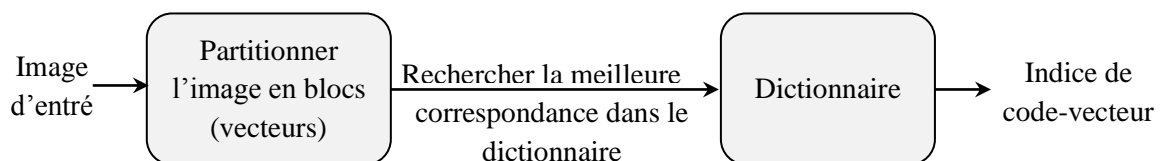
### 2.4.1 Algorithmes non-basés sur une transformée

Les algorithmes non basés sur une transformée incluent la compression fractales et la quantification vectorielle.

#### 2.4.1.1 Quantification vectorielle

La quantification vectorielle (QV) est une méthode conventionnelle de compression de données [Gresho 1991]. Elle peut être vue comme un mappage d'un grand ensemble de vecteurs dans un petit sous-ensemble de code-vecteurs appelé « le dictionnaire ». La QV permet de prendre en compte la corrélation entre les échantillons et de coder de manière efficace des échantillons représentés par plusieurs valeurs (ou vecteurs).

Comme le montre le schéma fonctionnel d'un quantificateur vectoriel décrit sur la figure 2.2, l'image est partitionnée en blocs de pixels, et chaque bloc est représenté par un vecteur  $x$ . Ce vecteur est comparé aux code-vecteurs figurant dans le dictionnaire de codes du côté codeur, ce qui donnera l'index de la meilleure correspondance de code-vecteur. Par conséquent, l'index stocké en moins de bits sera transmis à la place du code-vecteur, permettant ainsi d'atteindre un taux de compression plus élevé.



**Figure 2.2.** Schéma fonctionnel d'un quantificateur vectoriel.

L'avantage de la QV par rapport aux autres types de quantificateurs est la simplicité du décodeur. Cependant, l'inconvénient de base de la QV réside dans sa complexité, qui augmente avec l'augmentation de la dimensionnalité vectorielle. Cette complexité peut réduire la vitesse de codage et augmenter la dissipation d'énergie, en particulier dans les applications soumises à des contraintes d'énergie, telles que le cas des réseaux de capteurs. Un autre inconvénient de la QV est lié à la conception d'un dictionnaire universel de codes pour une grande base de données d'images, ce qui nécessite une taille mémoire importante avec également un grand nombre d'accès mémoire [Mammeri 2012].

#### **2.4.1.2 Compression fractales**

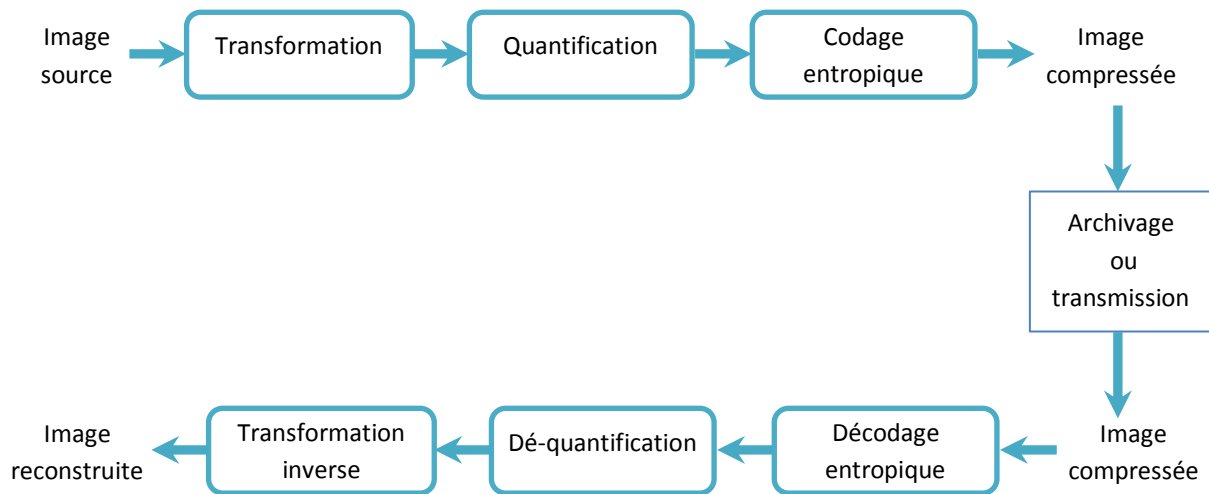
La compression fractale est une technique de compression avec pertes basée sur la théorie des fractales [Lu 1997], selon laquelle une image peut être décrite par un ensemble de fractales. L'idée derrière cette technique est que toute image est un ensemble fini de transformations géométriques (rotations, agrandissements, réductions) qui s'applique aux sous-ensembles de motifs identiques et de tailles variables qui la composent. La compression fractale consiste alors à remplacer l'image d'entrée par une série de formules mathématiques qui permettent de la recomposer dans sa totalité. L'efficacité de l'opération de compression est donc proportionnelle à l'importance des propriétés géométriques de l'image. Plus celles-ci sont nombreuses, moins le nombre de formules mathématiques sera conséquent.

La compression fractale d'images permet d'atteindre des taux de compression élevés avec une bonne qualité de l'image reconstruite. De plus, le processus de décodage est simple, qui est considéré comme une simple interprétation des codes fractals. Néanmoins, cette compression présente un inconvénient majeur lié principalement au processus de codage, qui est extrêmement intensif en calcul et consomme beaucoup de temps. Par conséquent, l'application de la compression fractale dans le domaine des réseaux de capteurs n'est pas bénéfique en termes de consommation d'énergie et peut réduire la durée de vie du réseau [Mammeri 2012].

#### **2.4.2 Algorithmes à base de transformée**

Les algorithmes de compression à base de transformée reposent généralement sur trois étapes, voir figure 2.3. La première étape est la transformation, ayant pour but de transformer le domaine de représentation de l'image. Elle permet de supprimer la redondance spatiale ou corrélation entre pixels voisins. La deuxième étape est la quantification, qui permet de réduire

l'espace de représentation des données. C'est le processus de perte réelle d'information, car la quantification implique la division et l'arrondi du résultat en un entier. La dernière étape est le codage entropique des données quantifiées. Elle permet d'organiser le train binaire en compressant les valeurs quantifiées sans altérer l'information.



**Figure 2.3.** Schéma général de compression/décompression à base de transformée.

#### – *Transformation*

Elle a comme but la dé-corrélation de l'information. Elle permet de supprimer la redondance entre pixels voisins, et de concentrer l'information pertinente de l'image sur un petit nombre de coefficients significatifs, ce qui permet d'optimiser les étapes ultérieures de quantification et de codage. Cette étape de traitement est réversible et sans perte.

Les techniques de compression reposant sur une étape de transformation figurent parmi les techniques les plus employées dans la compression des données et en particulier dans la compression d'images. Elles permettent d'atteindre des taux de compression élevés tout en préservant une bonne qualité de l'image reconstruite. Les transformées les plus populaires utilisées par des standards internationaux pour la compression des images sont : la transformée en cosinus discrète (DCT) [Ahmed 1974] employée dans JPEG et la transformée en ondelettes discrète (DWT) [Mallat 1989] utilisée dans JPEG2000.

#### – *Quantification*

Dans le schéma de compression, le processus de quantification est le principal responsable de la dégradation de l'image. C'est lors de cette étape que l'image va être réellement compressée. La quantification est souvent utilisée dans la pratique et elle est basée

sur le principe que les coefficients de basses fréquences font la plus importante contribution à la qualité de bloc reconstruit. Seuls ces coefficients, dont les magnitudes dépassent un seuil sont importants et tous ceux qui restent peuvent être écartés de la reconstruction de l'image.

La quantification peut s'effectuer échantillon par échantillon, on parle alors de quantification scalaire, ou bloc par bloc, dans ce cas la quantification est dite vectorielle. Cette dernière consiste à traiter conjointement  $n$  échantillons [Guillemot 2004].

L'élaboration d'un algorithme de quantification doit être conçue à partir de considérations visuelles, sur la base de la réaction du système visuel humain (HVS) aux variations de la luminance<sup>2</sup> et de la chrominance<sup>3</sup>. Il est à noter que le système visuel humain est davantage sensible aux fréquences basses qu'aux fréquences hautes, et que les hautes fréquences sont moins importantes que les basses fréquences lors de la reconstruction de l'image. Dans ce cas, les éléments jugés peu significatifs visuellement sont plus grossièrement quantifiés en utilisant un pas de quantification plus grand.

#### – *Codage entropique*

Le concept du codage entropique se fonde sur la théorie de l'information développée par Shannon. La stratégie consiste à exploiter l'analogie entre la probabilité d'un événement et la quantité d'information à transmettre. Le codage entropique permet d'encoder les symboles en leur attribuant un mot binaire. La longueur de ce mot est dépendante de la fréquence d'apparition du symbole dans le flux de données. Le principe d'un tel codage est d'encoder les symboles en utilisant des codes courts pour les événements souvent présents et des codes plus longs pour les occurrences rares.

Le codage entropique repose sur une procédure de codage à longueur variable, qui assigne des mots de longueur variable aux valeurs possibles, de façon telle que les symboles hautement probables soient associées à des mots courts du code, et les symboles de faible probabilité auront un code long. Ceci permet donc en principe de réduire la longueur moyenne des mots du code [Gosselin 2002]. La méthode pour générer de tels codes repose sur le principe du préfixe unique : aucun mot du code ne doit être le préfixe d'un autre mot. Par exemple, le code « 0, 10, 110, 111 » est un code préfixe alors que le code « 0, 10, 101,

---

<sup>2</sup> Luminance : est la sensation visuelle qu'on a de la lumière (intensité de la lumière).

<sup>3</sup> Chrominance : est un signal combiné comportant les trois informations de base : Rouge, Vert, Bleu. Elle est tout simplement la variation des couleurs

0101 » ne l'est pas, car « 0 » est le préfixe du mot « 0101 » et « 10 » est le préfixe de « 101 » [Guillemot 2004].

Parmi les codeurs entropiques les plus célèbres, on peut citer, le codage de Huffman [Huffman 1952], le codage arithmétique [Rissanen 1979] ou bien encore le codage de Golomb [Golomb 1966].

#### 2.4.2.1 Méthodes basées sur la transformée en cosinus discrète

La transformée en cosinus discrète (DCT) [Ahmed 1974] est la technique de transformation la plus largement utilisée en traitement de signal, qui a pour but de transformer le domaine de représentation de l'image. Elle permet de passer l'information de l'image du domaine spatial en une représentation identique dans le domaine fréquentiel. L'image est décomposée en plusieurs blocs, et pour chaque bloc, la DCT est exprimé mathématiquement comme une somme de fonctions cosinus oscillant à différentes fréquences.

Étant donné que la DCT est une transformation mathématique complexe. Elle consomme le plus d'énergie dans un schéma de compression basé sur la DCT (plus de 60% du coût total de l'algorithme JPEG [Taylor 2002]). De nombreux travaux visant à réduire sa complexité algorithmique ont été proposés dans la littérature. Certains de ces travaux, qui sont utiles pour le cas des réseaux de capteurs peuvent être classés comme suit :

- *Approche par factorisation de la matrice* : cette approche consiste à réécrire la transformée DCT sous forme matricielle et à factoriser la décomposition de manière à réduire le nombre de multiplications scalaires nécessaires. Une brève description de cette approche avec certains algorithmes de référence est rapportée dans [Makkaoui 2012].
- *Utilisation de la DCT à virgule fixe au lieu de la DCT à virgule flottante* : Comparée à la DCT à virgule flottante, la DCT à virgule fixe présente une efficacité énergétique plus importante. Cela peut justifier le choix possible de la DCT à virgule fixe dans le cas des réseaux de capteurs [Mammeri 2012].
- *Conversion des opérations gourmandes en énergie telles que les multiplications en des opérations simples* : En effet, le DCT peut être implémenté en utilisant uniquement que des opérations d'additions et de décalages. Cependant, cette

implémentation n'est qu'une approximation et elle ne fournit pas les mêmes précisions que la DCT réelle. Dans certaines applications, une simple approximation de la DCT peut fournir des résultats significatifs à une faible complexité arithmétique. Ainsi, les techniques d'approximation pour la DCT deviennent de plus en plus populaires. De telles approximations peuvent réduire les exigences de calcul de la DCT, conduisant à des implémentations rapides et à faible consommation d'énergie, tout en assurant une précision numérique adéquate. Une liste des algorithmes de l'état de l'art de la DCT approximative est documentée dans [Sejdic 2018].

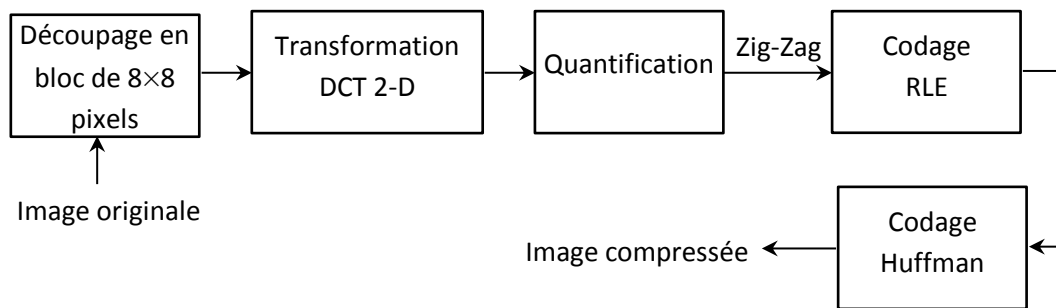
- *Approche zonale* : L'idée derrière cette technique est de réduire la complexité de la DCT en ne calculant qu'un sous-ensemble de tous les coefficients DCT, c'est-à-dire, ceux qui sont susceptibles d'être significatifs (composantes de basse fréquence).
- *Classe d'algorithmes conçue pour l'implémentation matérielle de la DCT* : Cette classe se concentre sur l'optimisation au niveau opérateurs arithmétique. Par conséquent, de nombreuses architectures sans multiplieurs ont été introduites pour l'implémentation efficace de la DCT. Toutes ces méthodes peuvent être classées en : design à base de mémoire morte (*Rom-based design*) [Slawecki 1992], design basé sur l'arithmétique distribuée (*DA-based design*) [White 1989], le nouveau design basé sur l'arithmétique distribuée (*NEDA-based design*) [Alam 2005] et le design basé sur calculateur numérique à rotation de coordonnées (*CORDIC-based design*) [Yu 2002].

Après avoir examiné l'idée principale derrière la DCT, nous présentons maintenant le standard JPEG [Wallace 1992], le schéma de compression bien connu basé sur la DCT, et nous discutons son éventuelle application dans le contexte des réseaux de capteurs.

#### – *Joint Photographic Experts Group (JPEG)*

La méthode de compression JPEG (*Joint Photographic Experts Group*) a été largement reconnue et elle est le plus souvent utilisée dans la compression d'image. La compression JPEG est une méthode de compression non conservative de l'image. L'algorithme a besoin de réaliser plusieurs étapes afin de pouvoir compresser une image. Premièrement, l'image de l'entrée est divisée en plusieurs blocs de taille fixe  $8 \times 8$  pixels. Ensuite la transformée de DCT bidimensionnelle (DCT 2-D) est appliquée sur chaque bloc afin de séparer les basses fréquences des hautes fréquences. Par la suite, chaque bloc DCT est quantifié uniformément. Le résultat de la quantification est réorganisé à nouveau d'une manière zigzag, des fréquences

basses aux fréquences hautes. Après cela, le codage RLE (*Run Length Encoding*) est appliqué pour réduire la longueur des séquences produites. Finalement, le processus de codage entropique réversible (tel que le codage Huffman ou arithmétique) est effectué sur les données quantifiées pour produire des mots de code de longueur fixes ou variables [Wallace 1992]. La chaîne de compression JPEG est illustrée sur la figure 2.4.



**Figure 2.4.** Étapes de la compression JPEG.

L'adoption de JPEG en tant qu'outils de compression dans les réseaux de capteurs n'est pas très bénéfique en termes de consommation d'énergie [Mammeri 2008a]. Cela est dû à la complexité de l'encodeur, et principalement à la complexité de l'étape de transformation DCT qui consomme seule plus de 60% de la totalité des exigences de calcul [Taylor 2002]. L'une des solutions les plus efficaces pour contourner ce problème consiste à réduire la complexité de la DCT. Cela conduit bien sûr à une réduction de la consommation d'énergie et, par conséquent, à une prolongation de la durée de vie du réseau, qui est un problème courant dans les réseaux de capteurs.

Les auteurs [Chiasserini 2002] réduisent le coût des étapes de DCT et de quantification de la chaîne de compression JPEG en faisant les opérations en arithmétique à virgule fixe plutôt qu'en virgule flottante. Selon l'exemple présenté dans [Chiasserini 2002], le codage d'une image QCIF en niveaux de gris avec une résolution de 1 bit par pixel (bpp) sur un microcontrôleur *StrongARM SA1110* avec JPEG (DCT à virgule fixe) a consommé 2.87 mJ. La même opération utilisant la DCT à virgule flottante nécessitait plus de 22 mJ.

Les auteurs [Lee 2009] ont suggérés une méthode qui détermine le nombre de bits optimal pour la partie entière et la partie fractionnaire des réels dans un processus de compression garantissant la précision requise. De plus, ils ont utilisés l'algorithme rapide de Loeffler–Ligtenberg–Moschytz (LLM) [Loeffler 1989], pour accélérer le calcul des

coefficients de la DCT. Il est à noter que les auteurs dans [Lee 2009] implémentent l'algorithme JPEG en arithmétique à virgule fixe au lieu de la représentation à virgule flottante pour des considérations énergétiques. Plusieurs expérimentations ont été effectuées en employant différents microcontrôleurs pour mesurer les économies d'énergie résultant du processus d'optimisation de la précision. Ces microcontrôleurs sont: Atmel ATmega128, TI MSP430, TI TMS320C64x et Analog Devices Blackfin ADSPBF533. La méthode suggérée surpasse le standard JPEG en termes de vitesse d'exécution et consommation d'énergie.

Comme rapporté dans [Kaddachi 2012], les auteurs ont proposé un schéma de compression reposant sur la structure conventionnelle à trois étapes <Transformation–Quantification–Codage entropique>. L'étape de transformation est basée sur la DCT rapide Cordic-Loeffler (CL-DCT) [Heyne 2006], utilisant que des opérations d'additions et de décalages pour le calcul des coefficients de la DCT. L'étape de quantification emploie la table de quantification recommandée par la norme JPEG [Wallace 1992]. L'étape de codage entropique est basée sur le parcours en zigzag, le codeur Golomb suivi par le codeur MQ. Le codeur MQ est une implémentation approximative du codage arithmétique adaptée pour les données binaires. L'utilisation du codage de Golomb suivi du codeur MQ au lieu du codage de Huffman ou encore le codage arithmétique est nettement rentable de point de vue complexité de calcul et utilisation de mémoire [Kaddachi 2012].

Les auteurs dans [Makkaoui 2010], ont proposé d'adapter l'algorithme JPEG aux contraintes imposées par les réseaux de capteurs ; en combinant une approche par factorisation de la matrice avec une approche par sélection zonale des coefficients. Cela revient à intégrer dans une chaîne de compression JPEG une DCT zonale rapide. Cette approche a permis de réduire le nombre de coefficients à calculer, à quantifier et à coder dans chaque bloc, entraînant mécaniquement des économies d'énergie sur toute la chaîne de compression. Dans la même veine, de nombreuses autres solutions dédiées aux réseaux de capteurs ont été proposées [Lecuire 2012, Kouadria 2013, Mechouek 2016, Araar 2017, Araar 2019].

Dans la section suivante, nous présentons la solution alternative à la DCT, c'est la transformée en ondelettes discrète (DWT), qui représente une technique prometteuse pour la compression d'images.

### 2.4.2.2 Méthodes basées sur la transformée en ondelette discrète

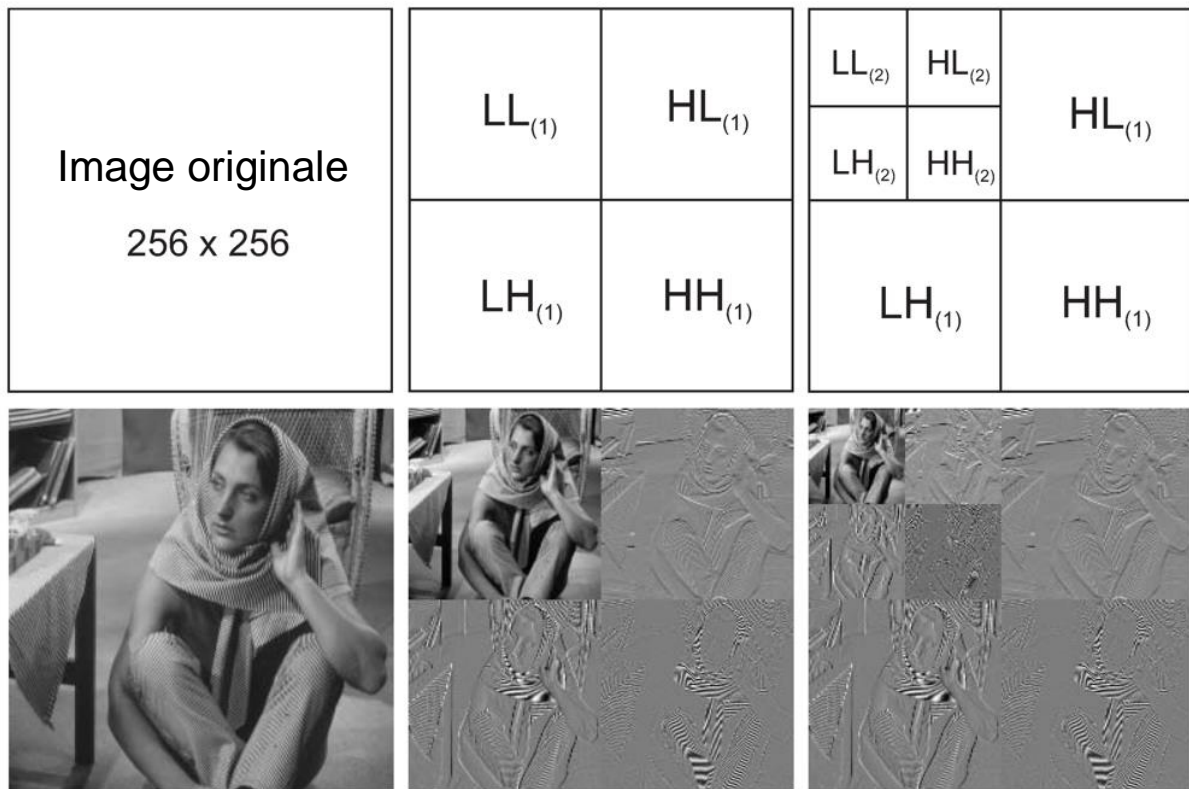
La transformée en ondelettes discrète (DWT) [Mallat 1989], est une opération qui décompose un signal en deux parties par projection sur un filtre passe-bas  $L$  et un filtre passe-haut  $H$ . La partie résultant du filtrage passe-bas est une approximation du signal d'origine à la nouvelle résolution (ou échelle), celle résultant du filtrage passe-haut représente les détails perdus entre les deux résolutions.

Puisque, en général, les ondelettes 2-D utilisées dans la compression d'images sont des fonctions séparables, son implémentation peut alors être obtenue en appliquant tout d'abord la DWT 1-D sur l'image d'entrée ligne par ligne pour produire les sous-bandes (sous-images)  $L$  et  $H$ , puis colonne par colonne pour en produire les quatre sous-bandes  $LL$ ,  $LH$ ,  $HL$  et  $HH$ . La sous-bande  $LL$  fournit une version de plus basse résolution (4 fois moins de pixels que l'image originale), tandis que les autres sous-bandes à savoir  $LH$ ,  $HL$  et  $HH$  représentent les détails perdus dans les directions horizontale, verticale et diagonale, respectivement. La DWT peut être réappliquée sur la sous-bande  $LL$  pour obtenir quatre sous-bandes de deuxième niveau :  $LL2$ ,  $LH2$ ,  $HL2$ ,  $HH2$ , et ainsi de suite, elle peut être réitérée plusieurs fois pour obtenir plusieurs niveaux de résolution. Les figures 2.5 et 2.6 montrent respectivement, la décomposition à deux niveaux d'une image  $256 \times 256$  par la DWT et les coefficients d'ondelette de chaque sous-bande.

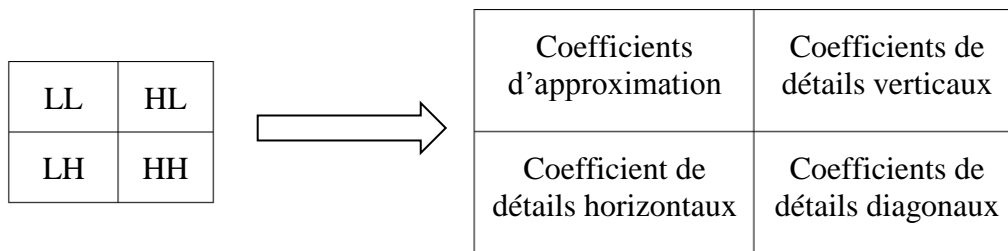
En raison de son utilité pour le compactage d'énergie du signal, une variété de schémas de compression d'images basés sur la DWT ont été développés. Citons à titre d'exemple, EBCOT, EZW et SPIHT. Dans ce qui suit, nous expliquerons brièvement les idées de base de ces systèmes, leurs avantages et inconvénients, ainsi que leurs applications dans les réseaux de capteurs.

#### – *Embedded Block Coding with Optimized Truncation (EBCOT)*

EBCOT est un algorithme de codage basé sur des blocs, dans lequel chaque sous-bande produite par la transformée DWT est divisée en petit blocs non chevauchants de coefficients DWT, appelés blocs de code. Chaque bloc de code est codé indépendamment, ce qui permet de générer un flux de bits hautement modulaires, plutôt que de générer un seul flux de bits représentant l'ensemble de l'image. Le standard de compression très populaire JPEG2000 adopte la technique de compression d'image EBCOT [Taubman 2000]. Le schéma de principe de JPEG2000 est illustré sur la figure 2.7.



**Figure 2.5.** Décomposition en ondelettes à deux niveaux de l'image Barbera.

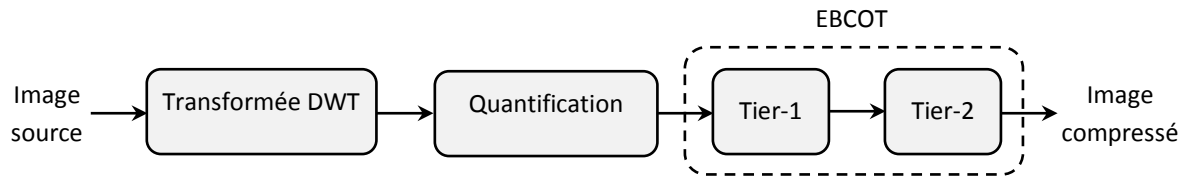


**Figure 2.6.** Les sous-bandes de la transformée en ondelettes et leurs contenus.

EBCOT est un algorithme de compression évolutif et efficace, résistant aux erreurs de transmission [Taubman 2000]. Néanmoins, l'algorithme EBCOT exige plus de ressources de calcul et de mémoire. Précisément, EBCOT utilise deux niveaux (Tier-1 et Tier-2) pour coder les informations, ce qui augmente le temps du traitement et implique une consommation d'énergie importante [Mammeri 2012].

Des techniques efficaces ont été suggérées pour améliorer la vitesse de codage et minimiser l'utilisation de la mémoire de l'algorithme EBCOT, par exemple [Lian 2003, Chiang 2004, Chang 2007]. Notre objectif n'étant pas de recenser toutes les techniques suggérées dans le domaine général de la compression d'images, mais, nous passons en revue

quelques travaux de recherche qui ont adopté EBCOT (ou JPEG2000) à des fins de compression dans les réseaux de capteurs.



**Figure 2.7.** Chaîne de compression JPEG2000.

Les auteurs dans [Yu 2004] proposent un schéma de compression basé sur JPEG2000 efficace en énergie pour le traitement et la transmission d'images. Dans le schéma suggéré, l'image d'entrée est d'abord codée dans un nombre optimal de couches (niveaux de qualité). En fonction des trois facteurs suivants: conditions estimées du canal de transmission, caractéristiques du contenu de l'image et contrainte de distorsion de bout en bout, le schéma détermine de manière adaptative les couches de qualité à transmettre. Les auteurs montrent par simulations que des économies d'énergie (jusqu'à 45%) peuvent être atteintes dans des conditions du canal relativement sévères.

Un autre schéma tenant compte de l'énergie pour la compression d'images dans les réseaux de capteurs est celui proposé dans [Wu 2004], où les auteurs ont introduit une technique qui emploie la norme JPEG2000 pour compresser des images capturées à partir des nœuds-caméras sans fil. Ils ont formulé le problème de transmission d'images comme un problème d'optimisation et ont proposé un algorithme heuristique appelé MTE (*Minimize Total Energy*). Pour une qualité d'image requise et des conditions de réseau données, les auteurs étudient l'algorithme heuristique pour sélectionner les paramètres optimaux d'un codeur à base d'ondelettes, tout en minimisant la dissipation totale d'énergie.

– ***Embedded zero-tree wavelet (EZW)***

L'auteur dans [Shapiro 1993] présente l'algorithme EZW, qui est l'un des premiers programmes implémenté pour montrer l'utilité du codage des images basé sur la transformée en ondelettes. L'algorithme EZW commence par effectuer la décomposition en ondelettes sur l'image d'entrée, ce qui permet sa décomposition en une série de coefficients d'ondelettes. Il suppose que si un coefficient d'ondelettes à un certain niveau de décomposition est inférieur à un seuil  $T$ , alors tous les coefficients de même orientation dans le même emplacement

spatial aux échelles de décomposition inférieures sont sans importance par rapport à  $T$ . Un coefficient d'ondelettes est dit significatif par rapport à  $T$  si sa valeur absolue est supérieure ou égale à  $T$ .

L'algorithme EZW est une procédure à passes multiples, où chaque passe implique deux étapes: passe dominante et passe subordonnée (ou passe de raffinement). Dans la passe dominante, la valeur initiale du seuil est choisie, contre laquelle tous les coefficients d'ondelettes sont comparés. Les coefficients sont ensuite codés selon leurs valeurs par rapport à la valeur fixée du seuil. Un coefficient d'ondelette est codé si sa valeur est supérieure ou égale au seuil  $T$ , sinon, il est traité comme indiqué dans [Shapiro 1993]. Les coefficients codés comme significatifs sont ensuite transférés vers la passe subordonnée. Dans cette passe, les coefficients significatifs trouvés dans la passe dominante sont quantifiés en utilisant une approche de quantification par approximation successive. Lorsque tous les coefficients d'ondelettes ont été balayés, la valeur du seuil est divisée par deux, et le processus de balayage est répété à nouveau pour ajouter plus de détails à l'image déjà encodée, jusqu'à ce que le débit binaire requis soit atteint.

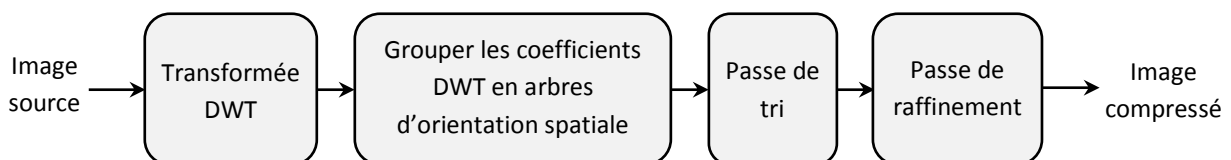
L'algorithme EZW est une méthode de compression simple et efficace. Ceci est obtenu grâce à une combinaison d'une transformation d'ondelette hiérarchique multi-résolution, d'un codage progressif des coefficients d'ondelettes, ainsi que d'une quantification par approximation successive. Le comportement intrinsèque de traitement progressif permet de terminer le processus de codage à tout moment, ce qui peut permettre en cas de réseaux de capteurs, des économies d'énergie dans le traitement et la communication. Cependant, EZW présente certains inconvénients. En fait, le nombre de passes nécessaires pour compresser une image affecte considérablement la qualité de l'image et l'énergie du nœud-caméra prenant en charge EZW. Une autre lacune d'EZW est liée à la mémoire requise pour stocker les coefficients d'ondelettes significatifs trouvés à chaque passe [Mammeri 2012].

Le seul travail de recherche adoptant EZW comme outil de compression pour les réseaux de capteurs est celui suggéré dans [Hamdi 2008], où les auteurs suggèrent une architecture de réseau de capteurs multimodale utilisant des capteurs acoustiques, électromagnétiques et visuels, ainsi qu'une épine dorsale de communication par satellite. Grâce à l'effort collaboratif de cette architecture, la position de la cible est reconnue et ses petits détails sont acquis à l'aide de capteurs visuels. À cet effet, l'algorithme de codage EZW est adapté aux exigences de réseaux de capteurs. Ceci est effectué en introduisant des

informations spatiales sur l'activité cible. L'algorithme EZW adapté a fourni des données de haute résolution pour les régions où une ou plusieurs intrusions ont été détectées et des données de basse résolution pour les régions restantes. Ce schéma a permis d'économiser de la bande passante, de l'énergie et des ressources de stockage.

– *Set Partitioning in Hierarchical Tree (SPIHT)*

L'algorithme SPIHT introduit dans [Saïd 1996] est une amélioration de l'algorithme EZW. SPIHT permet d'atteindre des performances de compression élevées en adoptant un algorithme de partitionnement d'ensemble et en explorant l'auto-similarité à différents niveaux d'une transformation en ondelettes. Contrairement à EZW, SPIHT maintient trois listes et quatre ensembles de coordonnées d'ondelettes, qui sont profondément expliquées dans [Saïd 1996]. Avec SPIHT, l'image est d'abord décomposée en une série de coefficients d'ondelettes. Ces coefficients sont ensuite groupés en ensembles, appelés arbres d'orientation spatiale. Après cela, les coefficients dans chaque arbre sont codés progressivement des plans de bits (*bit-planes*) les plus significatifs aux plans de bits les moins significatifs, en commençant par les coefficients avec la plus grande importance. Comme EZW, l'algorithme SPIHT implique deux passes de codage: passe de tri et passe de raffinement. La passe de tri recherche les arbres nuls (*zero-trees*) et trie les coefficients important et sans importance par rapport à un seuil donné  $T$ . La passe de raffinement envoie les bits de précision des coefficients significatifs. Après une passe de tri et une autre de raffinement, qui peuvent être considérées comme une seule passe de balayage, le seuil  $T$  est divisé par deux et le processus de codage est répété jusqu'à ce que le débit binaire attendu soit atteint. La figure 2.8 montre un schéma grossier des étapes de la compression SPIHT.



**Figure 2.8.** Etapes de la compression SPIHT.

SPIHT permet d'obtenir un flux binaire de sortie très compact, avec également un débit binaire bas que son prédécesseur EZW sans ajouter de codeur entropique, d'où son efficacité en termes de complexité de calcul. De plus, il utilise un schéma de partitionnement en sous-ensemble dans la passe de tri pour réduire le nombre de comparaisons, ce qui diminue également la complexité de calcul de l'algorithme. Enfin, le mode progressif de SPIHT

permet l'interruption du processus de codage/décodage à n'importe quelle étape de la compression [Said 1996]. Malgré ces avantages, SPIHT présente les lacunes suivantes, en particulier dans les applications limitée par énergie. Il nécessite une grande mémoire pour stocker les informations de codage. De plus, le filtre à ondelettes utilisé dans SPIHT est basé sur l'algorithme de Mallat [Mallat 1989], ce qui entraîne de grands calculs. Sur des réseaux non fiables, SPIHT est vulnérable à la perte de paquets, ce qui requiert l'utilisation d'un schéma de correction d'erreurs approprié [Mammeri 2012].

De nombreuses tentatives pour adapter l'algorithme SPIHT aux contraintes particulières des réseaux de capteurs ont été suggérées dans la littérature. Nous commençons par la méthode de compression proposée dans [Wu 2003], où l'idée de base est tirée de l'observation suivante. Il est plus efficace d'envoyer un flux binaire très long en petits fragments ou rafales (*bursts*) décomposés que sa transmission comme un bloc entier. C'est pourquoi, le schéma suggéré dans [Wu 2003] utilise une stratégie de décomposition basée sur les ondelettes pour créer plusieurs encodages du flux binaire des images qui sont envoyés en petites rafales. Les coefficients d'ondelettes sont groupés en plusieurs arbres et codés séparément à l'aide de l'algorithme SPIHT. Une méthode de protection contre les erreurs de transmission est également adoptée afin de lutter contre les erreurs du canal. Les résultats expérimentaux montrent que ce schéma présente une bonne efficacité énergétique en transmission.

Un autre travail incorporant SPIHT comme outil de compression est présenté dans [Chew 2009]. Les auteurs utilisent une technique de traitement par bandes où une image est divisée en plusieurs bandes qui seront encodées séparément. Premièrement, une bande d'image est chargée et décomposée en ondelettes par la transformée DWT. Après cela, les coefficients d'ondelettes sont calculés puis tamponnés dans un tampon de bandes pour l'encodage SPIHT. A la fin de la compression, le train binaire généré est transmis. Les auteurs ont montré que leur algorithme est capable de réduire considérablement la complexité et qu'il nécessite une quantité de mémoire beaucoup plus faible pour le traitement et la mise en mémoire tampon par rapport au codage SPIHT traditionnel, ce qui le rend approprié pour une implémentation dans des environnements sévèrement contraints en ressources tels que les réseaux de capteurs.

## **2.5 Conclusion**

Dans ce chapitre, nous avons passé en revue les techniques de codage ou compression d'images dans le domaine générale de la compression d'images. Deux paradigmes majeurs de compression d'images sont discutés; la compression avec pertes, appelée communément compression destructrice, et la compression sans perte, où la compression est un processus réversible qui n'entraîne aucune perte d'informations. Nous avons présentés brièvement les idées de base derrière ces approches, leurs avantages et inconvénients, ainsi que leurs applications dans le contexte des réseaux de capteurs.

Pour le paradigme de compression avec pertes, nous avons considéré deux classes d'algorithmes, ceux basés sur une transformée (DCT et DWT) et ceux qui ne sont pas à base de transformation (compression fractale et QV). Tout au long de la revue de la littérature, nous avons constaté que les algorithmes à base de transformation sont généralement préférés à ceux qui ne sont pas à base de transformation en termes d'efficacité énergétique. Cela est dû au fait que l'encodeur est moins complexe, ce qui justifie que leurs applications dans le domaine des réseaux de capteurs est très utiles.

## Deuxième partie

### **II. Contributions : Vers la compression d'images efficace en énergie dans les réseaux de capteurs sans fil**



# Chapitre 3

## Chapitre 3 : Propriétés et approches de réduction de la DCT

### Sommaire

---

<b>3.1</b>	<b>Introduction</b> .....	<b>53</b>
<b>3.2</b>	<b>Transformée en cosinus discrète</b> .....	<b>54</b>
3.2.1	DCT unidimensionnelle .....	54
3.2.2	DCT bidimensionnelle .....	58
<b>3.3</b>	<b>Propriétés de la DCT 2-D</b> .....	<b>62</b>
3.3.1	Dé-corrélation .....	63
3.3.2	Compactage d'énergie.....	65
3.3.3	Séparabilité .....	65
3.3.4	Symétrie .....	67
3.3.5	Orthogonalité .....	68
3.3.6	Normalisation .....	68
3.3.7	Orthonormalité.....	69
<b>3.4</b>	<b>Approche de réduction de la DCT</b> .....	<b>69</b>
3.4.1	DCT rapide.....	69
3.4.2	DCT zonale .....	75
<b>3.5</b>	<b>Conclusion</b> .....	<b>76</b>

---

### 3.1 Introduction

La théorie de la transformation joue un rôle clé dans le traitement de signal, et elle a été largement utilisée dans le codage d'images et de vidéo. L'adoption de fonctions de transformation pour coder les données de pixel repose sur la prémisse générale que les pixels adjacents présentent un niveau significatif de corrélation spatiale. En conséquence, la plupart des méthodes de codage d'images et de vidéo numériques qui ont été proposées tirent profit de ces fonctions de transformation pour mapper les données spatiales corrélées en un ensemble de coefficients de domaine de transformation moins corrélés. Certaines de ces fonctions de transformation les plus couramment utilisées pour le codage d'images et de vidéo sont : la transformée de Karhunen-Loève (KLT), la transformée de Fourier discrète (DFT), la transformée en cosinus discrète (DCT), la transformée en sinus discrète (DST) et la transformée en ondelettes discrète (DWT) [Mallat 1989, Rao 2001, Britanak 2007, Fugal 2009].

La KLT est la plus efficace, en termes de compactage d'énergie. Les fonctions de base de cette transformation sont obtenues à partir des propriétés statistiques des données de pixels. En conséquence, elle donne lieu à des performances optimales en termes de compactage d'énergie. Cependant, puisque le noyau de transformation de cette transformée dépend des données de l'image en cours de traitement, il ne peut pas être calculé en utilisant une forme de multiplication matricielle rapide, basée sur un noyau matriciel séparable et pré-calculé. Elle nécessite également une mise à jour continue des coefficients du noyau de transformation dans le décodeur, impliquant ainsi une diminution ultérieure du taux de compression. De plus, pour le codage par blocs, la dérivation du noyau de base correspondant à chaque bloc d'image introduit un effort de calcul supplémentaire, ce qui est un problème important dans la plupart des applications de codage d'images et de vidéo actuelles [Roma 2011].

Par conséquent, d'autres transformations moins efficaces mais indépendantes de l'image ont été préférées. Parmi eux, la DFT se caractérise par une définition séparable et symétrique. Contrairement à la KLT, elle est représentée par des fonctions de base fixées et présente également de bonnes caractéristiques de dé-corrélation et de compactage d'énergie. Cependant, la DFT est définie dans le domaine complexe et donc donne lieu à la fois à la magnitude et aux composantes de phase pour chaque échantillon. De plus, la périodicité implicite de la DFT introduit des discontinuités aux limites, ce qui se traduit par un contenu

haute-fréquence significatif [Blimm 1993]. C'est pourquoi, les transformées discrètes caractérisées par des fonctions de base plus souples ont été préférées. En particulier, la sortie fournie par la DCT conduit généralement à des niveaux d'efficacité de compactage assez proches des performances optimales fournies par la KLT [Roma 2011].

Au cours des dernières décennies, la DCT a émergée comme la transformation la plus largement adoptée dans de nombreuses normes de codage d'images et de vidéo. Dans ce chapitre, les concepts mathématiques de base de la DCT seront introduits. Puis, les propriétés les plus importantes de la DCT concernant la compression des données sont brièvement décrites. Enfin, une description des approches de réduction de la complexité de calcul de la DCT est donnée.

## 3.2 Transformée en cosinus discrète

La DCT constitue un outil puissant dans le traitement du signal depuis sa première introduction par [Ahmed 1974]. Elle appartient à une classe d'outils mathématiques qui comprend la fameuse transformée de Fourier DFT, ayant comme fonction de base la transformation d'un signal d'une représentation à une autre. Plus précisément, elle transforme un signal d'un domaine spatial vers l'espace fréquentiel.

La DCT est la technique de transformation la plus utilisée dans le traitement du signal et la compression des données. Il s'agit d'une transformation orthogonale, qui possède un ensemble fixe de fonctions de base, un algorithme efficace pour son calcul et de bonnes propriétés de compactage d'énergie et de réduction de corrélation. Comme les autres transformations, la DCT tente de dé-corréler les données d'un signal. Il existe plusieurs variantes de la DCT avec des définitions et des propriétés légèrement modifiées, comme la DCT de type I, II, III, IV, V-VIII [Britanak 2007, Roma 2011]. Parmi les différentes versions de DCT, les types II et III (DCT inverse) ont reçu beaucoup d'attention dans le traitement numérique du signal.

Dans ce qui suit, nous passons en revue la transformée DCT pour les deux cas : unidimensionnelle et bidimensionnelle.

### 3.2.1 DCT unidimensionnelle

La définition formelle de la DCT unidimensionnel (1-D) à  $N$ -points de  $N$  échantillons d'entrée  $x(0), \dots, x(N - 1)$  est donnée par :

$$y(u) = \sqrt{\frac{2}{N}} \alpha(u) \sum_{n=0}^{N-1} x(n) \cos\left(\frac{\pi(2n+1)u}{2N}\right) \quad (3.1)$$

avec  $u = 0, 1, \dots, N-1$ , et  $\alpha(u)$  est défini comme suit:

$$\alpha(u) = \begin{cases} \frac{1}{\sqrt{2}} & \text{si } u = 0 \\ 1 & \text{autrement} \end{cases} \quad (3.2)$$

De même, la DCT unidimensionnel inverse (IDCT 1-D) est définie comme donnée dans l'équation suivante :

$$x(n) = \sqrt{\frac{2}{N}} \sum_{u=0}^{N-1} \alpha(u) y(u) \cos\left(\frac{\pi(2n+1)u}{2N}\right) \quad (3.3)$$

La DCT unidimensionnel à  $N$ -points présentée dans l'équation (3.1) peut être exprimée sous forme matricielle comme :

$$y = C \cdot x, \quad (3.4)$$

où  $x = [x_0, x_1, \dots, x_{N-1}]^T$  est un vecteur d'entrée de taille  $N \times 1$ , et  $C$  est une matrice  $N \times N$  appelée matrice noyau de la transformation, dont les vecteurs de base sont des cosinus échantillonnés, donnée par :

$$C = \begin{bmatrix} c_{0,0} & c_{0,1} & \dots & c_{0,N-1} \\ c_{1,0} & c_{1,1} & \dots & c_{1,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ c_{N-1,0} & c_{N-1,1} & \dots & c_{N-1,N-1} \end{bmatrix}, \quad (3.5)$$

avec  $c_{i,j} = \sqrt{\frac{2}{N}} \alpha(i) \cos\left[\frac{\pi(2j+1)i}{2N}\right]$ , pour  $i, j = 0, 1, \dots, N-1$ .

La DCT inverse dans l'équation (3.3) peut être aussi formulée comme une multiplication matricielle selon :

$$x = C^T \cdot y, \quad (3.6)$$

où  $y = [y_0, y_1, \dots, y_{N-1}]^T$  est un vecteur de taille  $N \times 1$  éléments, et  $C^T$  est une matrice  $N \times N$  dont les éléments sont les mêmes que ceux définis dans l'équation (3.5), mais transposés.

### 3.2.1.1 Fonctions de base de la DCT 1-D

Dans les applications de compression d'images et de codage vidéo, la DCT à 8-points est couramment utilisée. Ainsi, pour  $N = 8$ , la matrice  $C$  dans l'équation (3.5) devient :

$$C = \begin{matrix} & \alpha_0 & \alpha_0 & \alpha_0 & \alpha_0 & \alpha_0 & \alpha_0 & \alpha_0 & \alpha_0 \\ \begin{matrix} \alpha_0 \\ \alpha_1 \cos\left(\frac{\pi}{16}\right) \\ \alpha_2 \cos\left(\frac{2\pi}{16}\right) \\ \alpha_3 \cos\left(\frac{3\pi}{16}\right) \\ \alpha_4 \cos\left(\frac{4\pi}{16}\right) \\ \alpha_5 \cos\left(\frac{5\pi}{16}\right) \\ \alpha_6 \cos\left(\frac{6\pi}{16}\right) \\ \alpha_7 \cos\left(\frac{7\pi}{16}\right) \end{matrix} & \begin{matrix} \alpha_0 \\ \alpha_1 \cos\left(\frac{3\pi}{16}\right) \\ \alpha_2 \cos\left(\frac{6\pi}{16}\right) \\ \alpha_3 \cos\left(\frac{7\pi}{16}\right) \\ -\alpha_4 \cos\left(\frac{4\pi}{16}\right) \\ -\alpha_5 \cos\left(\frac{\pi}{16}\right) \\ -\alpha_6 \cos\left(\frac{2\pi}{16}\right) \\ -\alpha_7 \cos\left(\frac{5\pi}{16}\right) \end{matrix} & \begin{matrix} \alpha_0 \\ \alpha_1 \cos\left(\frac{5\pi}{16}\right) \\ -\alpha_2 \cos\left(\frac{6\pi}{16}\right) \\ -\alpha_3 \cos\left(\frac{\pi}{16}\right) \\ -\alpha_4 \cos\left(\frac{4\pi}{16}\right) \\ \alpha_5 \cos\left(\frac{7\pi}{16}\right) \\ \alpha_6 \cos\left(\frac{2\pi}{16}\right) \\ \alpha_7 \cos\left(\frac{3\pi}{16}\right) \end{matrix} & \begin{matrix} \alpha_0 \\ \alpha_1 \cos\left(\frac{7\pi}{16}\right) \\ -\alpha_2 \cos\left(\frac{2\pi}{16}\right) \\ -\alpha_3 \cos\left(\frac{5\pi}{16}\right) \\ \alpha_4 \cos\left(\frac{4\pi}{16}\right) \\ \alpha_5 \cos\left(\frac{3\pi}{16}\right) \\ -\alpha_6 \cos\left(\frac{6\pi}{16}\right) \\ -\alpha_7 \cos\left(\frac{\pi}{16}\right) \end{matrix} & \begin{matrix} \alpha_0 \\ -\alpha_1 \cos\left(\frac{7\pi}{16}\right) \\ -\alpha_2 \cos\left(\frac{2\pi}{16}\right) \\ \alpha_3 \cos\left(\frac{5\pi}{16}\right) \\ \alpha_4 \cos\left(\frac{4\pi}{16}\right) \\ -\alpha_5 \cos\left(\frac{3\pi}{16}\right) \\ -\alpha_6 \cos\left(\frac{6\pi}{16}\right) \\ \alpha_7 \cos\left(\frac{\pi}{16}\right) \end{matrix} & \begin{matrix} \alpha_0 \\ -\alpha_1 \cos\left(\frac{5\pi}{16}\right) \\ -\alpha_2 \cos\left(\frac{6\pi}{16}\right) \\ \alpha_3 \cos\left(\frac{\pi}{16}\right) \\ -\alpha_4 \cos\left(\frac{4\pi}{16}\right) \\ -\alpha_5 \cos\left(\frac{7\pi}{16}\right) \\ \alpha_6 \cos\left(\frac{2\pi}{16}\right) \\ -\alpha_7 \cos\left(\frac{3\pi}{16}\right) \end{matrix} & \begin{matrix} \alpha_0 \\ -\alpha_1 \cos\left(\frac{3\pi}{16}\right) \\ \alpha_2 \cos\left(\frac{6\pi}{16}\right) \\ \alpha_3 \cos\left(\frac{7\pi}{16}\right) \\ -\alpha_4 \cos\left(\frac{4\pi}{16}\right) \\ \alpha_5 \cos\left(\frac{\pi}{16}\right) \\ -\alpha_6 \cos\left(\frac{2\pi}{16}\right) \\ \alpha_7 \cos\left(\frac{5\pi}{16}\right) \end{matrix} & \begin{matrix} \alpha_0 \\ -\alpha_1 \cos\left(\frac{\pi}{16}\right) \\ \alpha_2 \cos\left(\frac{2\pi}{16}\right) \\ -\alpha_3 \cos\left(\frac{3\pi}{16}\right) \\ \alpha_4 \cos\left(\frac{4\pi}{16}\right) \\ -\alpha_5 \cos\left(\frac{5\pi}{16}\right) \\ \alpha_6 \cos\left(\frac{6\pi}{16}\right) \\ -\alpha_7 \cos\left(\frac{7\pi}{16}\right) \end{matrix} \end{matrix} \\
 = \begin{matrix} \begin{matrix} 0.3536 \\ 0.4904 \\ 0.4619 \\ 0.4157 \\ 0.3536 \\ 0.2778 \\ 0.1913 \\ 0.0975 \end{matrix} & \begin{matrix} 0.3536 \\ 0.4157 \\ 0.1913 \\ -0.0957 \\ -0.3536 \\ -0.4904 \\ -0.4619 \\ -0.2778 \end{matrix} & \begin{matrix} 0.3536 \\ 0.2778 \\ -0.1913 \\ -0.4904 \\ -0.3536 \\ 0.0975 \\ 0.4619 \\ 0.41571 \end{matrix} & \begin{matrix} 0.3536 \\ 0.0975 \\ -0.4619 \\ -0.2778 \\ 0.3536 \\ 0.4157 \\ -0.1913 \\ -0.4904 \end{matrix} & \begin{matrix} 0.3536 \\ -0.0975 \\ -0.4619 \\ 0.2778 \\ 0.3536 \\ -0.4157 \\ -0.1913 \\ 0.4904 \end{matrix} & \begin{matrix} 0.3536 \\ -0.2778 \\ -0.1913 \\ 0.4904 \\ -0.3536 \\ -0.0975 \\ 0.4619 \\ -0.4157 \end{matrix} & \begin{matrix} 0.3536 \\ -0.4157 \\ 0.1913 \\ 0.0975 \\ -0.3536 \\ 0.4904 \\ -0.4619 \\ 0.2778 \end{matrix} & \begin{matrix} 0.3536 \\ -0.4904 \\ 0.4619 \\ -0.4157 \\ -0.3536 \\ -0.2778 \\ 0.1913 \\ -0.0975 \end{matrix} \end{matrix} \quad (3.7)$$

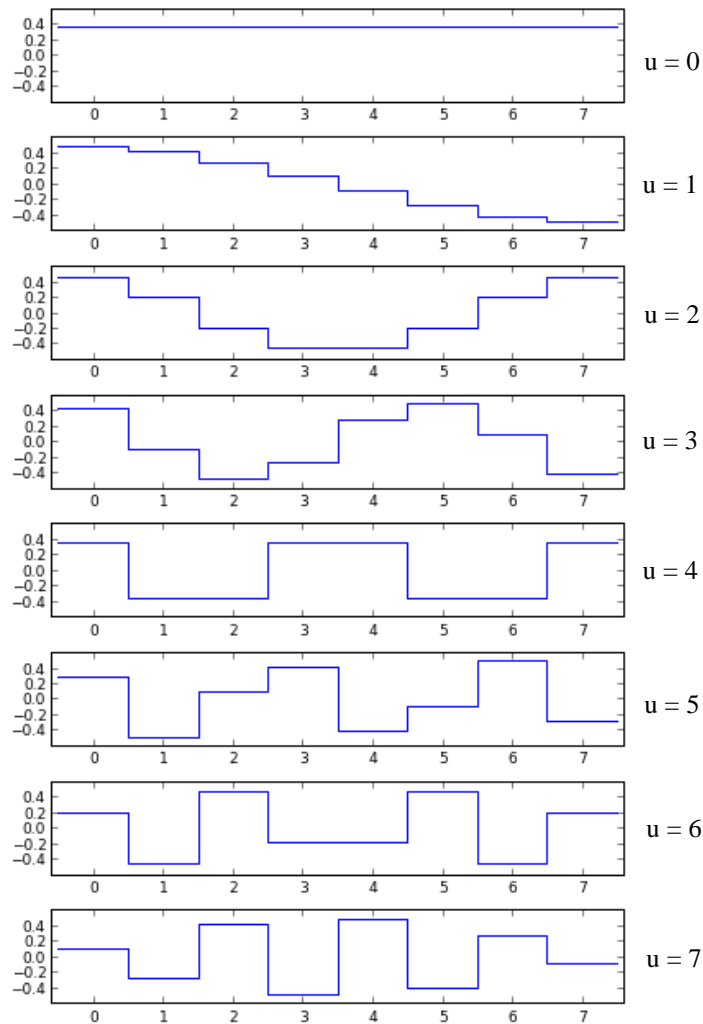
Laissons-nous examiner de près la DCT 1-D à 8-points. Un signal électrique d'amplitude constante est appelé signal DC (*direct current*). Un exemple courant est la batterie qui transporte 1.5 ou 9 volts de courant continu. Un signal électrique qui change périodiquement d'amplitude à une certaine fréquence est appelé signal AC (*alternative current*).

La plupart des signaux réels sont des signaux complexes. Cependant, tout signal peut être exprimé comme une somme de plusieurs signaux qui sont des formes d'onde sinusoïdale ou cosinoïdale de différentes amplitudes et fréquences. Les termes DC et AC, provenant de l'électrotechnique, sont utilisés pour décrire ces composantes d'un signal composé, généralement, d'une composante DC et de plusieurs composantes AC [Ze-Nian 2014].

Si une fonction cosinus est utilisée, le processus de détermination des amplitudes des composantes AC et DC du signal est appelé une transformation en cosinus, et les indices entiers en font une transformation en cosinus discrète [Ze-Nian 2014]. Ainsi, lorsque  $u = 0$ , l'équation (3.1) donne le coefficient DC; quand  $u = 1$ , ou 2, ..., jusqu'à 7, elle donne le premier ou le deuxième, ... etc., jusqu'au septième coefficient AC.

En bref, le rôle de la DCT est de décomposer le signal original en une composante DC et plusieurs composantes AC, chacune avec une fréquence particulière. Le rôle de l'IDCT est de reconstruire (recomposer) le signal. La DCT et l'IDCT utilisent le même ensemble de fonctions cosinus; on les appelle **fonctions de base** [Pennebaker 1993]. La figure 3.1

suivante montre la famille de huit fonctions de base de la DCT 1-D associées à la matrice  $C$  donnée dans l'équation (3.7).



**Figure 3.1.** Fonctions de base de la DCT 1-D ( $N = 8$ ).

Conformément à notre observation précédente, la première forme d'onde en haut ( $u = 0$ ) est la composante continue (DC) du signal, tandis que toutes les autres formes d'onde ( $u = 1, 2, \dots, 7$ ) reflètent les différentes composantes AC du signal variant à différentes fréquences.

Toutes ces fonctions de base sont orthogonales. Ainsi, la multiplication entre n'importe quelle paire de ces formes d'onde donne une valeur égale à zéro, tandis que la multiplication de l'une de ces formes d'onde avec elle-même donne une valeur constante. En conséquence, selon la définition orthogonale, ces formes d'onde sont dites indépendantes: aucune des fonctions de base ne peut être représentée comme une combinaison des autres fonctions de base [Strang 1999].

### 3.2.2 DCT bidimensionnelle

La DCT bidimensionnelle (2-D) a été adoptée dans plusieurs standards de codage d'images et de vidéos [Bhaskaran 1997], tels que JPEG [Wallace 1992] pour la compression d'images, dans les normes H.261 [ITU-T 1990], H.263 [ITU-T 1995], H.264/AVC [ITU-T 2003, Wiegand 2003] et H.265/HEVC [Pourazad 2012, ITU-T 2013] pour les applications de visioconférence, dans MPEG [Gall 1992] pour le stockage et la diffusion vidéo, ...etc. Le but de la DCT 2-D dans les standards de codage d'images et de vidéo est de réduire la redondance spatiale dans les images ou les frames vidéo, permettant ainsi de les coder en utilisant moins de bits.

Outre l'utilisation de la DCT 2-D comme partie intégrante des standards de codage d'images et de vidéos, elle est aussi largement utilisée comme méthode d'extraction de caractéristiques ou de réduction de dimensionnalité dans les applications de reconnaissance de formes [Sanderson 2003, Er 2005, Jadhav 2008, Liu 2008], dans le tatouage d'images et la dissimulation de données [Alturki 2007, Wong 2007, Qi 2008, Choi 2008], et dans diverses applications de traitement d'images [See 2007, Krupinski 2007, Abe 2007].

Soit  $X(n, m)$  un bloc d'entrée de taille  $N \times N$ . La DCT 2-D d'un signal bidimensionnel  $X(n, m)$  est défini comme la matrice de sortie de domaine de transformation de taille  $N \times N$  donnée par l'équation suivante :

$$Y(u, v) = \frac{2}{N} \alpha(u) \alpha(v) \sum_{n=0}^{N-1} \sum_{m=0}^{N-1} X(n, m) \cos\left(\frac{\pi(2n+1)u}{2N}\right) \cos\left(\frac{\pi(2m+1)v}{2N}\right) \quad (3.8)$$

pour  $u, v = 0, 1, \dots, N-1$

où les facteurs d'échelle  $\alpha(u)$  et  $\alpha(v)$  sont égaux à :

$$\alpha(u), \alpha(v) = \begin{cases} \frac{1}{\sqrt{2}} & \text{si } u, v = 0 \\ 1 & \text{autrement} \end{cases} \quad (3.9)$$

De même, la DCT bidimensionnelle inverse (IDCT 2-D) de  $Y(u, v)$  est défini comme :

$$X(n, m) = \frac{2}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} \alpha(u) \alpha(v) Y(u, v) \cos\left(\frac{\pi(2n+1)u}{2N}\right) \cos\left(\frac{\pi(2m+1)v}{2N}\right) \quad (3.10)$$

pour  $n, m = 0, 1, \dots, N-1$ .

La DCT est particulièrement adaptée aux signaux stationnaires. Or, la plupart des signaux réels tels que les sons ou les images ont des caractéristiques non stationnaires<sup>4</sup>. Pour traiter ces types de signaux transitoires, on peut découper le signal en plusieurs morceaux sur lesquels sera appliquée une transformée ; on fait ici en quelque sorte l'hypothèse que le signal est « **stationnaire par morceaux** ». C'est la méthode employée en particulier dans le standard de compression JPEG avec la transformée en cosinus par bloc. Toutefois, même si cette stratégie donne de bons résultats, elle présente l'inconvénient de créer des artéfacts (effets de bloc) dus au découpage en blocs [Guillemot 2004].

L'application de la DCT sur un bloc  $8 \times 8$ , donne 64 coefficients. Le premier coefficient DCT  $Y(0,0)$ , est le coefficient DC. Ce coefficient correspond à la valeur moyenne de la séquence d'entrée, et représente les détails les plus bruts du bloc (plus basse fréquence spatiale). Les coefficients restants sont appelés coefficients AC. Les coefficients AC représentent les détails les plus fins du bloc (fréquences spatiales plus élevées). Cela peut être démontré par un exemple. Prenons-nous un bloc  $8 \times 8$  à partir de l'image Lena, dont les intensités de pixels sont représentées sur la figure 3.2. Puis, calculons-nous la DCT en utilisant l'équation (3.8). Les valeurs du bloc transformé sont représentées sur la figure 3.3.

Il est ressort clairement de la figure 3.3 que la plupart des coefficients transformés ont de très petites valeurs et que seulement quelques coefficients ont des amplitudes plus élevées. La composante DC trouvée en position (0,0) transporte une information plus utile que les autres composantes hautes fréquences. Plus on s'éloigne de la composante DC, on constate non seulement que les coefficients ont tendance à avoir de faibles valeurs mais aussi, qu'ils deviennent moins importants pour la description de l'image. Ce qui signifie qu'en effectuant la transformée DCT sur un bloc de pixels, nous concentrons l'énergie de l'image dans les coefficients situés en haut et à gauche de bloc de sortie. Cela justifie la capacité de compactage d'énergie de la DCT.

---

<sup>4</sup> Les signaux non stationnaires sont des signaux qui ne se répètent que rarement et se manifestent par des oscillations évoluant au cours du temps.

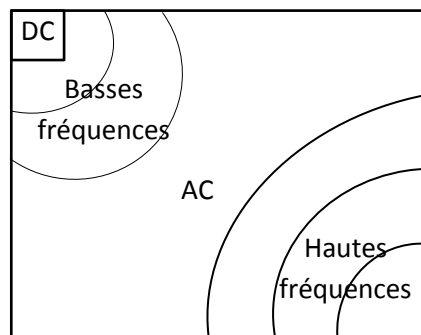
52	55	61	66	71	61	64	73
63	59	66	90	109	585	69	72
62	59	68	113	144	104	66	73
63	58	71	122	154	106	70	69
67	61	68	104	126	88	68	70
79	65	60	70	77	68	58	75
85	71	64	59	55	61	65	83
87	79	69	58	65	76	78	94

**Figure 3.2.** Un bloc d'échantillons  $8 \times 8$  de l'image Lena.

DC →	609	-29	-62	25	55	-20	-1	3
	7	-21	-62	9	11	-7	-6	6
	-46	8	77	-25	-30	10	7	-5
AC →	-50	13	35	-15	-9	6	0	3
	11	-8	-13	-2	-1	1	-4	1
	-10	1	3	-3	-1	0	2	-1
	-4	-1	2	-1	2	-3	1	-2
	-1	-1	-1	-2	-1	-1	0	-1

**Figure 3.3.** Coefficients DCT pour le bloc  $8 \times 8$  de l'image Lena.

On peut représenter la distribution des fréquences de la DCT d'un bloc  $8 \times 8$  comme illustrée sur la figure 3.4. L'intégralité de l'information de l'image se trouve dans les basses fréquences. Par contre, les détails de l'image se localisent en hautes fréquences.



**Figure 3.4.** Répartition des fréquences dans un bloc DCT  $8 \times 8$ .

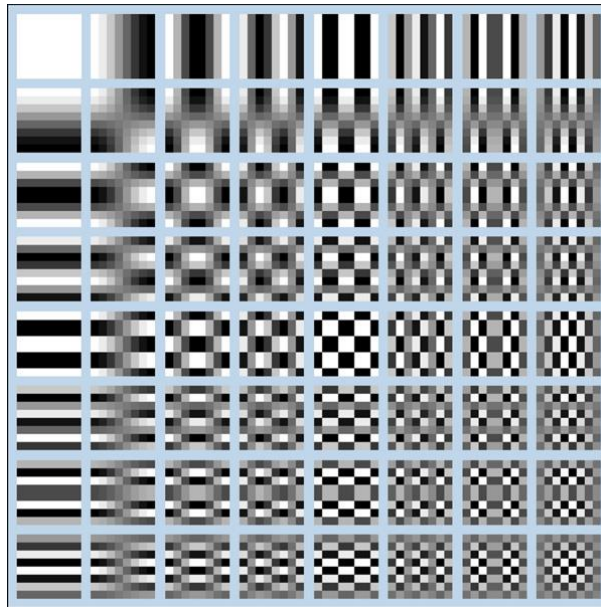
### 3.2.2.1 Fonctions de base de la DCT 2-D

La transformation DCT 2-D est en fait une décomposition de l'image en plusieurs sous-images de base souvent appelées fonctions de base. Sur la figure 3.5, il est représenté

l'ensemble de 64 fonctions de base correspondant à la DCT 2-D, pour  $N = 8$ . Ces fonctions de base peuvent être générées soit: (1) en multipliant les fonctions de base unidimensionnelles orientées horizontalement (montrés sur la figure 3.1) par un ensemble orienté verticalement des mêmes fonctions, où l'ensemble de fonctions de base orienté horizontalement représente les fréquences horizontales, et l'autre ensemble de fonctions de base représente les fréquences verticales [Pennebaker 1993] (2) à l'aide du noyau de transformation bidimensionnelle donné par [Réf. 01] :

$$G(n, m, u, v) = \frac{2}{N} \alpha(u) \alpha(v) \cos\left(\frac{\pi(2n+1)u}{2N}\right) \cos\left(\frac{\pi(2m+1)v}{2N}\right). \quad (3.11)$$

Pour chaque valeur de  $u$  et  $v$  ( $u, v = 0, 1, \dots, N-1$ ), on obtient une fonction de base de taille  $N \times N$  en calculant l'équation (3.11) sur  $n$  et  $m$  ( $n, m = 0, 1, \dots, N-1$ ). Pour un bloc de  $8 \times 8$ , nous obtenons donc 64 fonctions de base chacune de taille  $8 \times 8$ .



**Figure 3.5.** Fonctions (sous-images) de base de la DCT 2-D ( $N = 8$ ). Le gris neutre représente zéro, le blanc représente les amplitudes positives et le noir les amplitude négative [Pennebaker 1993].

On peut noter que les fonctions de base de la DCT 2-D présentent une augmentation progressive de la fréquence dans les deux directions horizontale et verticale. Les fréquences horizontales augmentent de gauche à droite et les fréquences verticales augmentent de haut en bas. Un cas particulier se produit avec la fonction de base en haut à gauche (composante

DC), qui résulte de la multiplication de la composante continue de la figure 3.1 avec sa transposition. Par convention, la composante DC des fonctions de base horizontales est à gauche, et la composante DC pour les fonctions de base verticales est en haut. Par conséquent, la ligne supérieure et la colonne de gauche présentent des variations d'intensité unidimensionnel, qui, si elles étaient tracées, seraient les mêmes que celles sur la figure 3.1 [Pennebaker 1993].

Pour calculer les coefficients DCT 2-D, nous utilisons les fonctions de base montrées sur la figure 3.5 ci-dessus, en faisant juste le produit matricielle de chacune de ces 64 fonctions de base avec un bloc  $8 \times 8$  pixels d'une image originale.

### 3.2.2.2 Méthode de calcul de la DCT 2-D

Il existe deux méthodes pour le calcul de la DCT 2-D : méthode directe et méthode de décomposition ligne-colonne.

#### – Méthode de DCT 2-D directe

La transformation DCT 2-D doit être appliquée directement sur le bloc de données d'entrée. Ainsi, pour un bloc de données de  $N \times N$  éléments, le calcul de la DCT 2-D originale nécessite  $N^4$  multiplications et  $N^2 \cdot (N^2 - 1)$  additions [Kesheh 1999].

#### – Méthode de décomposition ligne-colonne

La méthode de décomposition ligne-colonne est une méthode basée sur la propriété de séparabilité du noyau de la DCT 2-D. Elle consiste à appliquer séquentiellement la DCT 1-D sur les lignes du bloc de données d'entrée, puis à transposer les résultats intermédiaires, suivi de l'application d'une autre DCT 1-D sur les colonnes du bloc de données transformées. Ainsi, pour un bloc de données  $N \times N$ , le calcul de la DCT 2-D requière seulement  $2N^3$  multiplications et  $2N^2 \cdot (N - 1)$  additions [Ling 2012].

## 3.3 Propriétés de la DCT 2-D

Plusieurs propriétés utiles peuvent être dérivées de la transformée DCT précédemment définie. Cette section présente un bref aperçu des propriétés les plus importantes et utiles de cette transformation [Khayam 2003, Roma 2011, Brahimi 2011, Kouadria 2014].

### 3.3.1 Dé-corrélation

Les données des pixels voisins présentent un niveau significatif de corrélation spatiale pour la plupart des images. Par conséquent, le principal objectif de la transformation d'image est la dé-corrélation des données de pixels afin de minimiser la redondance spatiale entre les pixels adjacents. Cela conduit à des coefficients transformés non corrélés, et qui peuvent être codés indépendamment afin d'obtenir le niveau de compression le plus élevé possible.

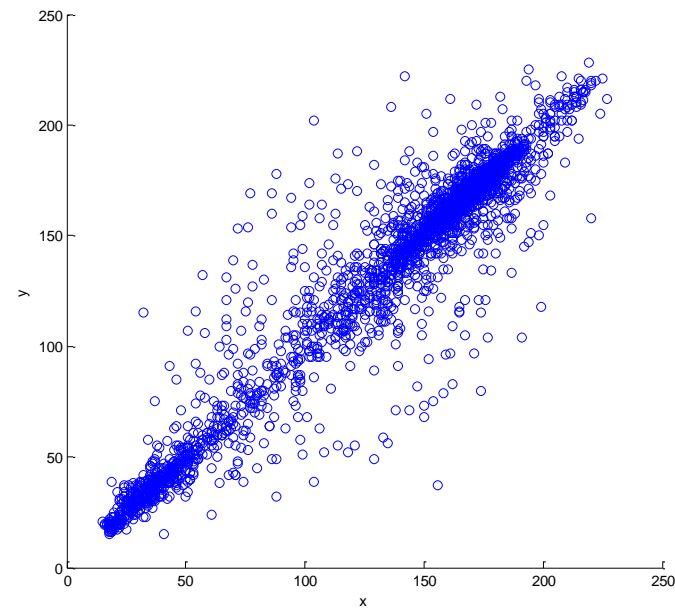
La figure 3.6 montre une image en niveaux de gris typique. L'image est de taille  $512 \times 512$  pixels, chaque valeur de niveau de gris du pixel étant représentée par une valeur sur 8 bits pour une plage de  $[0 - 255]$ . Cette image particulière est couramment utilisée dans les évaluations, et elle est souvent appelée l'image Boat. Même avec un grand degré de détail dans de nombreuses régions, la valeur du niveau de gris d'un pixel donné tend à être similaire aux valeurs de ses pixels voisins [Rao 2001]. Pour illustrer cette relation, nous avons tracés les valeurs de niveau de gris de paires de pixels adjacents avant et après application de la DCT, comme montré sur la figure 3.7.



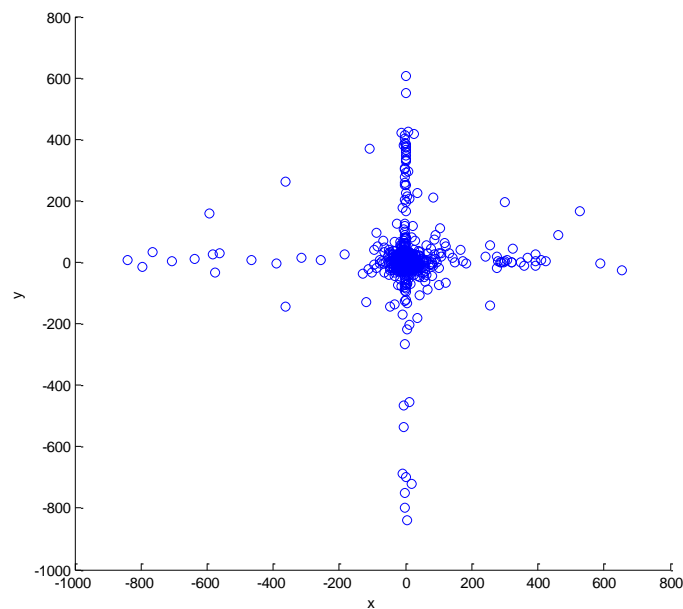
**Figure 3.6.** Exemple d'une image en niveau de gris typique (Boat).

Chaque point (petit cercle sur la figure 3.7-(a)) représente un pixel dans l'image, la coordonnée  $x$  étant sa valeur de niveau de gris et la coordonnée  $y$  étant la valeur de niveau de gris de son voisin de droite. La forte relation diagonale autour de la ligne  $x = y$  montre clairement la forte corrélation entre les pixels voisins [Rao 2001], voir figure 3.7-(a). Cela

veut dire que la valeur d'un pixel donné peut être prédite à partir de ses voisins correspondants.



(a)



(b)

**Figure 3.7.** Diagramme de distribution des paires de valeurs de pixels adjacents. (a) avant la DCT. (b) après la DCT.

Maintenant, que se passerait-il après application de la transformée DCT sur l'image de la figure 3.6? Le résultat est illustré sur la figure 3.7-(b). Les deux composantes sont maintenant dé-corrélées, c'est-à-dire que la connaissance de la valeur de la première

composante n'aide pas à estimer la valeur de la seconde. Par conséquent, il peut être déduit que le DCT présente d'excellente propriété de dé-corrélation.

### 3.3.2 Compactage d'énergie

Une autre propriété importante de la transformée DCT est lié à sa capacité à concentrer la plupart de l'énergie du signal dans les coefficients de basse fréquence. L'une des conséquences de cette caractéristique est que l'élimination de certains coefficients de haute fréquence (par quantification et troncature) entraîne généralement une perte marginale de l'énergie du signal, entraînant un niveau de distorsion minimal [Blinn 1993].

La DCT présente un excellent compactage d'énergie pour les images hautement corrélées. Des études ont montré que les performances de la propriété de compactage d'énergie de la DCT se rapprochent de l'optimalité à mesure que la corrélation d'image s'approche, c'est-à-dire que la DCT permet de fournir une dé-corrélation (presque) optimale pour ces images [Clark 1985].

Pour mesurer l'efficacité de compactage d'énergie (*Energy Packing Efficiency – EPE*) de la DCT, nous devons calculer l'équation (3.12). L'efficacité de compactage d'énergie a été proposée pour la première fois par Kitajima pour la transformée WHT [Kitajima 1976], puis étendue à d'autres transformées orthogonales discrètes. Elle est définie comme la portion d'énergie contenue dans les premiers  $m$  de  $n$  coefficients de transformation [Rao 1990].

$$EPE(m) = \frac{\sum_{k=0}^m y^2(k)}{\sum_{k=0}^{N-1} y^2(k)}, \quad m = 0, 1, \dots, N - 1 \quad (3.12)$$

Des exemples de la propriété de compactage d'énergie de la DCT de certaines images standard sont illustrés sur la figure 3.8 [Khayam 2003].

### 3.3.3 Séparabilité

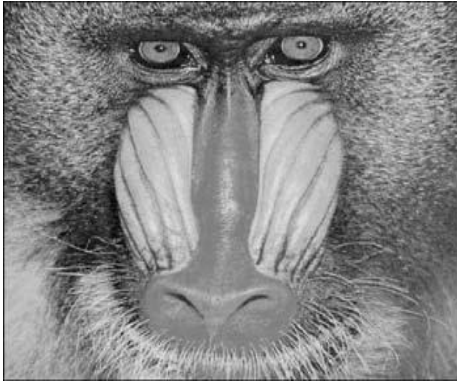
La propriété des fonctions séparables peut être appliquée pour simplifier le calcul de la DCT 2-D. La transformée DCT donnée dans l'équation (3.8) peut être réécrite comme :

$$Y(u, v) = \sqrt{\frac{2}{N}} \alpha(u) \sum_{n=0}^{N-1} \left[ \sqrt{\frac{2}{N}} \alpha(v) \sum_{m=0}^{N-1} X(n, m) \cos\left(\frac{\pi(2m+1)v}{2N}\right) \right] \cos\left(\frac{\pi(2n+1)u}{2N}\right) \quad (3.13)$$

pour  $u, v = 0, 1, \dots, N - 1$ .

De la même manière, la DCT 2-D inverse dans l'équation (3.10) peut être exprimée comme donnée par l'équation (3.14) :

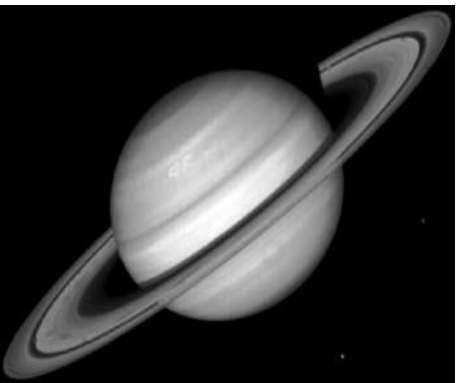
$$X(u, v) = \sqrt{\frac{2}{N}} \alpha(u) \sum_{u=0}^{N-1} \left[ \sqrt{\frac{2}{N}} \alpha(v) \sum_{v=0}^{N-1} Y(u, v) \cos\left(\frac{\pi(2m+1)v}{2N}\right) \right] \cos\left(\frac{\pi(2n+1)u}{2N}\right) \quad (3.14)$$



(a)



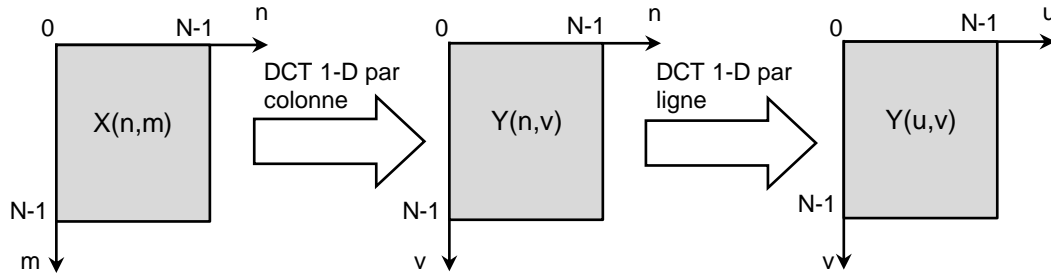
(b)



(c)

**Figure 3.8.** (a) Baboon et sa DCT. (b) Child et sa DCT. (c) Saturn et sa DCT [[Khayam 2003](#)].

Le principal avantage de la propriété de séparabilité est que  $Y(u, v)$  ou  $X(n, m)$  peut être obtenu en deux étapes par applications successives de la transformée de DCT 1-D ou de son inverse. Le principe de cette idée est illustré graphiquement sur la figure 3.9.



**Figure 3.9.** Calcul de la DCT 2-D en utilisant deux transformations successives de DCT 1-D.

Tout d'abord, nous calculons une fonction intermédiaire  $Y(n, v)$  en effectuant une transformée DCT 1-D sur chaque colonne. De cette façon, nous sommes passés à l'espace fréquentiel pour les colonnes, mais pas pour les lignes :

$$Y(n, v) = \sqrt{\frac{2}{N}} \alpha(v) \sum_{m=0}^{N-1} X(n, m) \cos\left(\frac{\pi(2m+1)v}{2N}\right) \quad (3.15)$$

Le résultat souhaité  $Y(u, v)$  est ensuite obtenu en effectuant une autre transformation le long de chaque ligne de  $Y(n, v)$ , comme indiqué dans l'équation ci-dessous.

$$Y(u, v) = \sqrt{\frac{2}{N}} \alpha(u) \sum_{n=0}^{N-1} Y(n, v) \cos\left(\frac{\pi(2n+1)u}{2N}\right) \quad (3.16)$$

Ceci est possible car les fonctions de base de la DCT 2-D sont séparables. Il est facile de voir que ce simple changement permet d'économiser beaucoup d'étapes arithmétiques. Le nombre d'itérations nécessaires est réduit de  $N^2$  à  $2N$  [Ze-Nian 2014]. Cela représente un gain important par rapport à une simple double boucle sur les indices comme pourrait le laisser suggérer la formule initiale de la DCT 2-D.

### 3.3.4 Symétrie

Un autre regard sur l'équation (3.8) indique que les opérations de lignes et de colonnes pour le calcul de la DCT 2-D sont fonctionnellement identiques. Une telle transformation est appelée une transformation symétrique. Une transformation séparable et symétrique peut être exprimée sous la forme [Radha 2003] :

$$Y = C.X.C^T, \quad (3.17)$$

où  $Y$  est la matrice transformée de sortie de taille  $N \times N$ ,  $X$  est la matrice  $N \times N$  d'entrée,  $C$  est la matrice noyau de la transformation donnée par l'équation (3.5), et le  $(.)^T$  désigne l'opération de transposition matricielle.

Cette propriété est d'une extrême importance car elle implique que la matrice noyau de la transformée peut être pré-calculée hors ligne, puis appliquée à l'image, ce qui permet d'améliorer grandement l'efficacité de calcul [Khayam 2003].

### 3.3.5 Orthogonalité

Les vecteurs de ligne et de colonne qui composent la matrice noyau de la transformation DCT définissent un ensemble de fonctions de base orthogonales.

Etant donnée la matrice noyau  $C$  définie dans l'équation (3.5). Cette matrice est dite orthogonale car tous les vecteurs colonnes  $c_i = [c_{0,i} \ c_{1,i} \ \dots \ c_{N-1,i}]^T$  remplit la relation suivante, notée par  $c_i \perp c_j$  :

$$c_0^T c_1 = c_1^T c_2 = \dots = c_i^T c_j = 0 \ \forall i \neq j \quad (3.18)$$

avec  $0 \leq i, j \leq N - 1$ .

En conséquence, cette relation implique également que :

$$C^T = C^{-1} \quad (3.19)$$

Des relations tout à fait similaires s'appliquent également aux vecteurs de lignes.

En vertu de l'orthogonalité, la transformée de la DCT 2-D inverse de l'équation (3.10) peut être facilement formulée en notation matricielle comme suit

$$X = C^T.Y.C \quad (3.20)$$

### 3.3.6 Normalisation

Chaque vecteur de colonne  $c_i$  de la matrice noyau de la DCT remplit la propriété suivante:

$$\|c_i\| = 1, \forall i: 0 \leq i \leq N - 1 \quad (3.21)$$

En conséquence, la matrice noyau est également dite normalisée.

### 3.3.7 Orthonormalité

Étant donné que chaque vecteur de colonne ou de ligne de la matrice noyau de la DCT est à la fois orthogonal et normalisé, cette matrice est également dite orthonormée, présentant ainsi les propriétés importantes suivantes:

$$C^T = C^{-1}, \quad C^T.C = I, \quad C.C^T = I \quad (3.22)$$

Ces relations conduisent à une conséquence assez importante, puisque l'opération d'inversion de matrice est réduite à une simple transposition matricielle, cela entraîne une réduction significative des coûts de calcul [Roma 2011].

## 3.4 Approche de réduction de la DCT

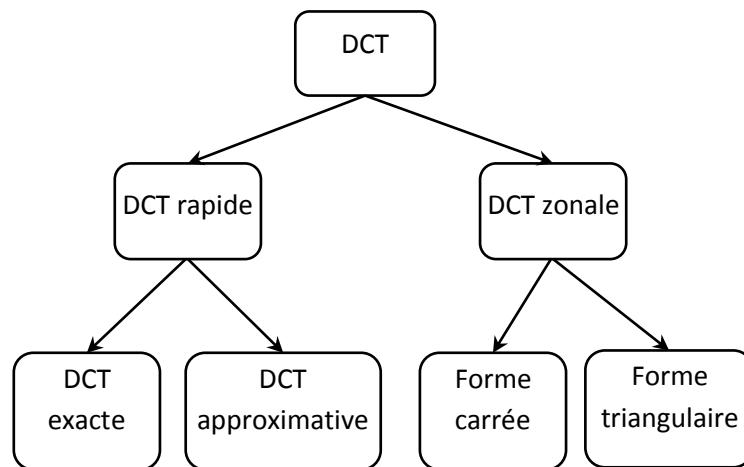
Après l'introduction de la DCT par Ahmed et al. en 1974 [Ahmed 1974], la conception des algorithmes de DCT efficaces a été un effort scientifique majeur chez la communauté des circuits, systèmes et traitement du signal avec d'énormes applications pour la compression d'images et vidéos. En raison de ces recherches intenses dans ce domaine, plusieurs techniques visant à réduire la complexité de calcul de la DCT ont été proposées dans la littérature. Une classification possible de ces techniques est la suivante, voir figure 3.10 :

- Approche visant à réduire le nombre d'opérations nécessaires pour calculer les coefficients DCT, appelée approche de **DCT rapide** ou approche par factorisation de la matrice.
- Approche permettant de réduire le nombre de coefficients DCT en ne calculant qu'un sous-ensemble de tous les coefficients, appelée **DCT zonale**, ou également DCT *pruning*.

### 3.4.1 DCT rapide

La DCT a toujours présentée une complexité de calcul notable, qui nécessite des algorithmes rapides dédiés pour son calcul efficace. En fait, il peut être facilement observé qu'un calcul direct de la DCT 1-D à 8-points (équation 3.1) nécessite 64 multiplications et 56 additions, tandis qu'une DCT 2-D (équation 3.8) nécessite jusqu'à 4096 multiplications et 4032 additions. Ce problème devient particulièrement préoccupant lorsque l'on s'attend à ce

qu'une énorme quantité de données soit traitée. Bien que ce coût de calcul puisse être réduit de manière triviale à 1024 multiplications et 896 additions en utilisant la propriété de séparabilité de la DCT, il représente toujours une charge de calcul importante pour la plupart des applications pratiques. Par conséquent, un nombre important d'algorithmes rapides pour le calcul de la DCT ont été développés au cours des dernières décennies. La plupart de ces approches exploitent largement les propriétés de la matrice noyau de la transformation  $C$  ou la précision finale du résultat de la transformation, afin de réduire le nombre d'opérations impliquées [Roma 2011].



**Figure 3.10.** Approches de réduction de la DCT.

### 3.4.1.1 DCT exacte

De nombreux algorithmes rapides pour le calcul de la DCT exacte sont archivés dans la littérature. Certains des algorithmes les plus populaires pour le calcul la DCT à 8-points avec leur complexité arithmétique sont présentés dans le tableau 3.1. Ces algorithmes utilisent l'arithmétique à valeur réelle, et ils sont basés sur des techniques de factorisation matricielle, permettant alors de réduire le nombre des opérations arithmétiques nécessaires pour le calcul de la DCT. En effet, ces algorithmes permettent d'atteindre des économies significatives dans le coût de calcul par rapport au calcul direct de la DCT.

Algorithme	DCT 1-D		DCT 2-D	
	Mult.	Add.	Mult.	Add.
DCT par définition [Ahmed 1974]	64	56	1024	896
Chen et al. [Chen1977]	16	26	256	416
Wang [Wang 1984]	13	29	208	464
Lee [Lee 1984]	12	29	192	464
Vetterli-Nussbaumer [Vetterli 1984]	12	29	192	464
Suehiro-Hatori [Suehiro 1986]	12	29	192	464
Hou [Hou 1987]	12	29	192	464
Loeffler et al. [Loeffler 1989]	11	29	176	464
Feig-Winograd [Feig 1992]	22	28	352	448
Jeong et al. [Jeong 1998]	12	28	192	448
Yuan et al. [Yuan 2006]	12	29	192	464

**Table 3.1.** Complexité de calcul des algorithmes rapides de la DCT 1-D et 2-D.

Tous les algorithmes rapides susmentionnés visent à calculer la DCT exacte et ils ont encore besoin de multiplications en virgule flottante, qui sont à la fois lentes aussi bien dans les implémentations logicielles que matérielles [Liang 2001]. De plus, des algorithmes ont atteint la borne inférieure théorique du nombre de multiplications requis (11 multiplications) pour la DCT à 8-points [Duhamel 1987, Loeffler 1989]. Par conséquent, il peut être irréaliste de s'attendre à ce que de nouveaux algorithmes rapide de la DCT exacte puissent offrir des gains de calcul spectaculaires. Ainsi, les progrès dans ce domaine se sont orientés vers des méthodes approximatives. Les algorithmes résultants sont beaucoup plus rapides que leurs prédécesseurs et ont donc de larges applications pratiques [Madanayake 2015, Sejdic 2018].

### 3.4.1.2 DCT approximative

Bien que les algorithmes rapides de la DCT exacte puissent réduire considérablement la complexité de calcul de la DCT, des opérations en virgule flottante sont encore nécessaires [Britanak 2007]. Malgré leur précision, les opérations en virgule flottante sont coûteuses en termes de complexité des circuits et de consommation d'énergie [Potluri 2014]. Par conséquent, la réduction du nombre d'opérations en virgule flottante est très recherchée dans un algorithme rapide. Une façon de contourner ce problème est au moyen de transformations approximatives.

Les méthodes approximatives offrent un autre moyen de réduire davantage la complexité de calcul de la DCT. Bien qu'elles ne calculent pas la DCT exactement, de telles approximations peuvent fournir des estimations significatives de la DCT au détriment de petites erreurs introduites à la précision obtenue [Cintra 2011b]. Un tel compromis est souvent acceptable, conduisant à des implémentations matérielles à haute vitesse et à faible consommation d'énergie, tout en assurant une précision numérique adéquate [Potluri 2014]. Dans ce sens, la littérature a été peuplée de méthodes approximatives pour le calcul efficace de la DCT à 8-points. Une revue de la littérature des méthodes de la DCT approximative se trouve dans [Sejdic 2018]. Les principales techniques incluent : la DCT signée classique – *signed DCT*– (SDCT) [Haweel 2001], la DCT approximative par Lengwehasatit–Ortega [Lengwehasatit 2004], la série d'algorithmes de Bouguezel–Ahmad–Swamy (BAS) [Bouguezel 2008a, Bouguezel 2008b, Bouguezel 2009, Bouguezel 2010, Bouguezel 2011, Bouguezel 2013], la DCT arrondie – *the rounded DCT* – (RDCT) [Cintra 2011b], la DCT arrondie modifiée – *the modified rounded DCT* – (MRDCT) [Bayer 2012], l'approximation pour l'imagerie radiofréquence [Potluri 2012] et la DCT approximative améliorée [Potluri 2014].

En effet, la DCT approximative peut remplacer la DCT exacte dans les implémentations matérielles et les calculs/traitements à haute vitesse, tout en ayant de faibles coûts matériels et de faibles besoins en énergie [Britanak 2007]. Effectivement, la DCT approximative a été envisagée pour des applications dans le traitement et la transmission de vidéo en temps réel [Kuo 2011, Saponara 2012], les systèmes de communication par satellite [Britanak 2007], les applications d'informatiques portables [Britanak 2007], les réseaux d'antennes à radiofréquence intelligentes [Potluri 2012], et les applications de réseaux de capteurs d'images sans fil [Lecuire 2012, Kouadria 2013, Mechouek 2016, Araar 2017, Araar 2019].

Généralement, la DCT approximative est une transformation  $\hat{C}$ , qui, selon une certaine métrique spécifiée telle que la proximité de la transformation par rapport à la DCT exacte ou les performances de codage [Tablada 2017], se comporte de manière similaire à la matrice noyau de la DCT exacte  $C$ . Une matrice d'approximation  $\hat{C}$  est généralement basée sur une matrice de transformation  $T$  (composante clé d'une DCT approximative) de faible complexité de calcul. Souvent, les éléments de la matrice de transformation  $T$  possèdent une complexité multiplicative nulle. Par exemple, cette propriété peut être satisfaite en restreignant les

entrées de la matrice  $T$  à l'ensemble des puissances de deux  $\{0, \pm 1, \pm 2, \pm 4, \pm 8, \dots\}$  [Madanayake 2015, Sejdic 2018]. En fait, les multiplications par de tels éléments sont triviales et ne nécessitent que des opérations de décalage de bits. De manière générale,  $\hat{C}$  est une matrice aux valeurs réelles qui se compose de deux composantes: une matrice de faible complexité et une matrice diagonale, qui peuvent être mise dans le format suivant:

$$[\text{matrice diagonale}] \times [\text{matrice de faible complexité}]$$

Dans ce qui suit, nous adoptons la terminologie suivante. Une matrice  $A$  est dite orthogonale si le produit  $A.A^T$  est une matrice diagonale. En particulier, si  $A.A^T$  est la matrice d'identité, alors  $A$  est orthonormée.

Etant donné une matrice de transformation de faible complexité  $T$ , il est possible de dériver une matrice orthonormée  $\hat{C}$  basée sur  $T$  au moyen de la décomposition polaire [Cintra 2011a, Seber 2008] :

$$\hat{C} = S.T, \quad (3.23)$$

où  $T$  est une matrice de faible complexité donnée par :

$$T = \begin{bmatrix} t_{0,0} & t_{0,1} & \dots & t_{0,N-1} \\ t_{1,0} & t_{1,1} & \dots & t_{1,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ t_{N-1,0} & t_{N-1,1} & \dots & t_{N-1,N-1} \end{bmatrix}, \quad (3.24)$$

avec  $t_{i,j}, \{i, j = 0, 1, \dots, N-1\}$  sont les entrée de la matrice  $T$ , et  $S$  est une matrice symétrique définie positive [Seber 2008] donnée par :

$$S = \sqrt{(T.T^T)^{-1}}, \quad (3.25)$$

où  $\sqrt{\cdot}$  désigne l'opération de racine carrée de la matrice. Etant orthonormée, ce type d'approximation satisfait  $\hat{C}^{-1} = \hat{C}^T$ . Par conséquent, nous avons

$$\hat{C}^{-1} = T^T.S \quad (3.26)$$

Par conséquent, la transformation inverse  $\hat{C}^{-1}$  hérite de la même complexité de calcul que la transformation directe  $\hat{C}$ .

Du point de vue de calcul, il est souhaitable que  $S$  soit une matrice diagonale. Dans ce cas, la complexité de calcul de  $\hat{C}$  est la même que celle de  $T$ , à l'exception des coefficients dans la matrice diagonale  $S$ . De plus, selon l'application considérée, même les constantes de  $S$  peuvent être ignorées en termes d'évaluation de la complexité de calcul. Cela se produit lorsque les constantes impliquées sont des multiplicandes triviaux, tels que les puissances de deux [Madanayake 2015, Sejdic 2018]. Une autre possibilité plus pratique de négliger la complexité de  $S$  apparaît quand elle peut être absorbée dans d'autre étape dans un algorithme de compression. C'est le cas de la compression de type JPEG, où l'étape de quantification est présente [Wallace 1992]. Ainsi, la matrice  $S$  peut être incorporée dans la matrice de quantification [Lengwehasatit 2004, Bouguezel 2008b, Cintra 2011b]. En termes de transformation inverse, il est également avantageux que  $S$  soit diagonal, car la complexité de  $\hat{C}^{-1}$  devient essentiellement celle de  $T^T$ .

Pour que  $S$  soit une matrice diagonale, il suffit que  $T$  satisfait à la condition d'orthogonalité [Madanayake 2015, Sejdic 2018]:

$$T \cdot T^T = D, \quad (3.27)$$

où  $D$  est une matrice diagonale.

Si l'équation (3.27) n'est pas satisfaite, alors  $S$  n'est pas une matrice diagonale et les propriétés avantageuses de la DCT approximative résultante sont en principe perdues. Dans ce cas, les éléments hors-diagonale contribuent à une augmentation de la complexité de calcul et l'absorption de la matrice  $S$  ne peut pas être facilement réalisée. Cependant, au prix de ne pas fournir une approximation orthogonale, on peut envisager d'approximer  $S$  elle-même en remplaçant les éléments hors diagonale de  $D$  par des zéros [Madanayake 2015, Sejdic 2018]. Ainsi, la matrice résultante  $\hat{S}$  est donnée par:

$$\hat{S} = \sqrt{\text{diag}[(T \cdot T^T)^{-1}]}, \quad (3.28)$$

où  $\text{diag}[\cdot]$  renvoie une matrice diagonale avec les éléments diagonaux formés à partir de son argument. Ainsi, l'approximation non-orthogonale est fournie par:

$$\tilde{C} = \hat{S} \cdot T \quad (3.29)$$

La matrice  $\tilde{C}$  peut-être une approximation significative si  $\hat{S}$  est, dans un certain sens, proche de  $S$ ; ou, alternativement, si  $T$  est presque orthogonale [Tablada 2017, Madanayake 2015, Sejdic 2018].

Du point de vue de conception algorithmique, proposer des approximations non-orthogonales peut être une tâche moins exigeante, car l'équation (3.27) n'est pas tenue d'être satisfaite [Madanayake 2015, Sejdic 2018]. Cependant, comme  $\tilde{C}$  n'est pas orthogonal, la transformation inverse doit être examinée avec prudence. En effet, la transformation inverse n'emploie pas directement la matrice de faible complexité  $T$  et est donnée par :

$$\tilde{C}^{-1} = T^{-1} \cdot \hat{S}^{-1} \quad (3.30)$$

Même si  $T$  est une matrice de faible complexité, il n'est pas garanti que  $T^{-1}$  soit également de faible complexité de calcul. Néanmoins, il est possible d'obtenir des approximations non-orthogonales dont les matrices de transformation directe et inverse ont une faible complexité de calcul. Deux exemples d'approximations non-orthogonales sont la SDCT [Haweel 2001] et l'approximation décrite dans [Bouguezal 2008a].

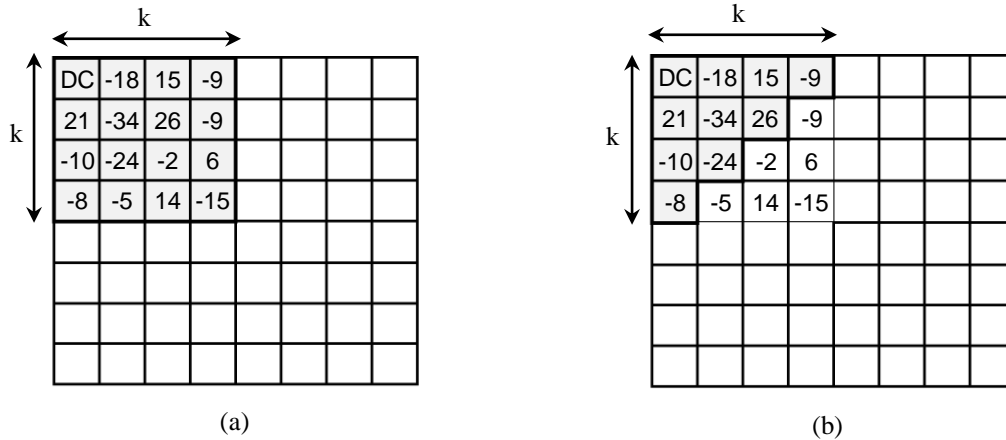
### 3.4.2 DCT zonale

L'approche de la DCT zonale, également connue dans la littérature par « DCT *pruning* », proposée la première fois pour la DFT [Markel 1971], puis pour la DCT [Wang 1991], est basée sur la prémisse que les coefficients de basse fréquence sont ceux qui transportent la plupart du signal (information importante) et doivent être conservés, tandis que ceux de haute fréquence peuvent être tronqués.

La DCT zonale consiste à calculer un sous-ensemble des  $N \times N$  coefficients DCT, en ne conservant que les coefficients inclus dans la zone carrée supérieure gauche de taille  $k \times k$ , dont le nombre de coefficients est  $k^2$ , où  $k < N$  (typiquement,  $N = 8$  dans le cas de JPEG), comme illustré sur la figure 3.11-(a). Il y a donc  $k^2$  coefficients DCT à calculer à la place de  $N^2$ . Les autres ne sont pas pris en compte et donc ne seront pas calculés. Par conséquent, des coûts de calcul peuvent être économisés car seulement un sous-ensemble des coefficients transformés est calculé.

La variable  $k$  est un paramètre qui délimite horizontalement et verticalement la zone de sélection des coefficients à calculer. Par exemple, dans le cas de la forme carrée, seulement  $k^2$

coefficients (16 coefficients pour  $k = 4$ ) doivent être calculés au lieu de 64 coefficients pour la DCT 2-D classique, voir figure 3.11-(a). Si au contraire, la forme triangulaire est adoptée, il y'a uniquement  $k \cdot (k + 1)/2$  coefficients à calculer (soit 10 coefficients pour  $k = 4$ ), voir figure 3.11-(b)).



**Figure 3.11.** Principe de la DCT zonale ( $k = 4$ ). (a) forme carrée. (b) forme triangulaire.

Dans la littérature, la DCT zonale de forme carrée est la plus usuelle, mais elle n'est pas la seule forme possible. Dans [Mammeri 2008b] par exemple, les auteurs considèrent une forme triangulaire dont la longueur des cotés est  $k$ . Cependant, dans un travail présenté dans [Makkaoui 2010], les auteurs ont montré que la DCT zonale utilisant une forme carrée offre un meilleur compromis énergie-distorsion que celle utilisant une forme triangulaire.

### 3.5 Conclusion

Dans ce chapitre, il a été présenté un aperçu général de la transformation DCT, avec un accent particulier sur la présentation des principales propriétés de la DCT 2-D. Cette transformation a été particulièrement adoptée dans plusieurs standards de compression d'images et vidéo, en raison de sa grande capacité à exploiter les redondances inter-pixels et de son excellente capacité de compacter efficacement la majeure partie de l'énergie du signal dans un nombre réduit de composants de fréquence.

Nous avons également discuté deux approches majeures qui servent à calculer la DCT avec une complexité réduite. L'approche de la DCT zonale, qui sert à réduire le nombre de coefficients DCT à calculer et l'approche de la DCT rapide qui permet de réduire le nombre d'opérations nécessaire pour le calcul des coefficients de la DCT.

L'approche de la DCT rapide, et l'approche de la DCT zonale présentées dans ce chapitre sont complémentaires et peuvent être combinées. Une telle combinaison est désignée comme étant une DCT zonale rapide. Cette dernière entraîne une réduction importante dans la complexité calculatoire de la DCT, et par conséquent une réduction dans la consommation d'énergie. Notre contribution qui sera développée dans le chapitre suivant, suit cette voie en proposant de combiner une méthode de sélection zonale des coefficients avec une méthode de DCT approximative.



# Chapitre 4

## Chapitre 4 : DCT zonale approximative et architecture matérielle proposées

### Sommaire

---

<b>4.1</b>	<b>Introduction .....</b>	<b>78</b>
<b>4.2</b>	<b>DCT zonale exacte et approximative.....</b>	<b>79</b>
<b>4.3</b>	<b>DCT zonale approximative proposée.....</b>	<b>81</b>
4.3.1	Transformée proposée.....	81
4.3.2	Évaluation de la complexité .....	84
<b>4.4</b>	<b>Évaluation des performances.....</b>	<b>86</b>
4.4.1	Temps d'exécution et consommation d'énergie .....	86
4.4.2	Mémoire requise.....	88
4.4.3	Compression d'images.....	89
<b>4.5</b>	<b>Architecture VLSI.....</b>	<b>93</b>
4.5.1	Architecture matérielle proposée .....	94
4.5.2	Implémentation sur FPGA.....	97
<b>4.6</b>	<b>Conclusion.....</b>	<b>98</b>

---

## 4.1 Introduction

En raison de l'augmentation rapide des plates-formes de capteurs dotées de capacités d'imagerie, la compression d'images efficace en énergie sur des réseaux de capteur sans fil devient de plus en plus importante. Dans de tels réseaux, la transmission d'images à travers des nœuds-capteurs limités en ressources se traduit par une surconsommation d'énergie et une utilisation accrue de la bande passante, outre les coûts additionnels pour l'acquisition, le traitement et le stockage de données au niveau du nœud-capteur. L'une des solutions les plus efficaces pour réduire la dissipation d'énergie du nœud-capteur et d'utiliser la bande passante de manière plus efficace consiste à recourir aux techniques de compression à la source. Habituellement, la compression d'images à la source est un choix idéal pour surmonter les goulots d'étranglement de la bande passante et de l'énergie.

Bien que les schémas classiques de compression d'images à base de transformation, tels que SPIHT, JPEG2000 ou encore JPEG, ont montré leur efficacité dans la compression d'images en termes de rapport débit-distorsion, ils ne sont pas éligibles dans le contexte des réseaux de capteurs, car ils exigent beaucoup de calculs et d'accès mémoire avec le risque que cela coûte plus d'énergie de compresser l'image que de la transmettre sans compression [Ferrigno 2005].

Les schémas de compression basés sur la transformée DWT tels que SPIHT et JPEG2000 sont coûteux en temps et en énergie par rapport à ceux basés sur la DCT (JPEG). Il a été rapporté dans [Ferrigno 2005] que le temps d'exécution de SPIHT et de JPEG2000 était respectivement, 2 et 6 fois plus long que JPEG. De plus, la DWT est coûteuse en termes d'utilisation de mémoire puisqu'elle s'applique typiquement sur l'image entière, contrairement à la DCT qui s'applique sur des blocs de petite taille, généralement, des blocs de  $8 \times 8$  pixels [Makkaoui 2012]. En conséquence, la mémoire requise pour stocker les résultats intermédiaires serait de grande dimension. Compte tenu de ce qui précède, nous nous intéressons essentiellement à la compression JPEG qui semble être une bonne candidate pour l'adapter aux contraintes imposées par les réseaux de capteur sans fil.

Au cours des dernières années, l'algorithme de compression JPEG s'est largement répandu, et il est le plus souvent utilisé dans la compression d'images. Néanmoins, l'adoption de cet algorithme comme outils de compression dans les réseaux de capteurs n'est pas très bénéfique en termes de consommation d'énergie [Mammeri 2008a, Lee 2009]. Cela est dû (1)

aux nœuds-capteurs qui sont soumis à des contraintes très strictes en termes de mémoire, de capacité de traitement et surtout d'énergie, rendant ainsi inefficace l'exécution de cet algorithme, et (2) à la complexité de l'encodeur JPEG, et principalement à la complexité de l'étape de transformation DCT qui consomme au moins 60% du coût total de la consommation en énergie [Taylor 2002]. Une façon de contourner ce problème consiste à réduire la complexité calculatoire de la DCT.

Récemment, certaines méthodes de DCT approximative qui visent à réduire la complexité de calcul de la DCT ont été proposées [Cintra 2011b, Bayer 2012, Bouguezal 2013, Potluri 2014, Tablada 2017, Jridi 2017, Oleivera 2018]. Bien qu'elles ne permettent pas d'offrir les mêmes précisions que les méthodes de DCT exacte, ces approximations peuvent fournir des estimations importantes de la DCT au détriment d'une légère augmentation de la distorsion de l'image compressée.

Afin de réduire davantage la complexité de calcul de la DCT, nous proposons de combiner une méthode de DCT approximative avec une approche zonale. Le but de la première est de réduire le coût de calcul de la DCT en employant que des opérations d'addition au lieu des opérations coûteuses de multiplication, tandis que la seconde vise à calculer uniquement les coefficients de basse fréquence les plus importants. Cela se traduit par une réduction dans la consommation d'énergie et, par conséquent, une prolongation de la durée de vie du réseau, qui est un problème courant dans les réseaux de capteurs sans fil.

L'objectif du présent chapitre est double: d'une part, introduire une nouvelle DCT zonale approximative présentant un coût de calcul extrêmement faible, seulement 10 additions sont nécessaires pour son calcul et, d'autre part, concevoir une architecture matérielle efficace et de faible complexité pour la DCT zonale proposée et les méthodes concurrentes.

## 4.2 DCT zonale exacte et approximative

L'approche de la DCT zonale consiste essentiellement à extraire de la matrice noyau  $C$  de l'équation (3.5) une sous-matrice qui vise à fournir des propriétés mathématiques similaires à celles de  $C$ . Généralement, seuls les coefficients  $k^2 < N^2$  qui retiennent plus d'énergie sont préservés. Mathématiquement, cela correspond à la conservation des  $k$  premières lignes de la matrice noyau  $C$ . De telle opération conduit à la nouvelle matrice  $C_k$  de  $k \times N$  entrées suivante :

$$C_k = \begin{bmatrix} c_{0,0} & c_{0,1} & \dots & c_{0,N-1} \\ c_{1,0} & c_{1,1} & \dots & c_{1,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ c_{k-1,0} & c_{k-1,1} & \dots & c_{k-1,N-1} \end{bmatrix}, \quad (4.1)$$

où  $0 < k \leq N - 1$ , et  $c_{i,j}$  sont les entrée de la matrice  $C_k$ .

Par conséquent, la DCT 2-D zonale exacte directe et inverse sont calculées comme suit :

$$\tilde{Y} = C_k \cdot X \cdot C_k^T \quad (4.2)$$

$$X = C_k^T \cdot \tilde{Y} \cdot C_k \quad (4.3)$$

La matrice résultante  $\tilde{Y}$  est une matrice carrée de taille  $k \times k$ . Comme il a été montré dans [Makkaoui 2010], le maintien des coefficients du domaine fréquentiel dans une forme carrée de taille  $k \times k$  conduit à un meilleur compromis énergie-distorsion par rapport à la forme triangulaire alternative. Par conséquent, la forme carrée sera considérée dans ce travail.

L'approche de la DCT zonale peut également être appliquée à la DCT approximative. En éliminant de la matrice de faible complexité  $T$  donnée dans l'équation (3.24), les lignes inférieures correspondant aux coefficients de haute fréquence. Ainsi, nous obtenons la matrice de transformation zonale de  $k \times N$  entrées suivante :

$$T_k = \begin{bmatrix} t_{0,0} & t_{0,1} & \dots & t_{0,N-1} \\ t_{1,0} & t_{1,1} & \dots & t_{1,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ t_{k-1,0} & t_{k-1,1} & \dots & t_{k-1,N-1} \end{bmatrix}, \quad (4.4)$$

avec  $t_{i,j}$  sont les entrée de la matrice  $T$ ,  $i = 0,1, \dots, k - 1$  et  $j = 0,1, \dots, N - 1$ .

Si  $T$  est orthogonale, alors  $T_k$  satisfait à la semi-orthogonalité [Abadir 2005]. La DCT 1-D zonale approximative  $k \times N$  de la transformation présentée dans l'équation (3.23) peut être exprimée comme suit:

$$\hat{C}_k = S_k \cdot T_k, \quad (4.5)$$

où  $S_k$  est une matrice diagonale  $k \times k$  donnée par :

$$S_k = \sqrt{(T_k \cdot T_k^T)^{-1}}. \quad (4.6)$$

Pour le cas 2-D, la DCT zonale approximative directe et inverse sont données respectivement par :

$$\tilde{Y} = \hat{C}_k \cdot X \cdot \hat{C}_k^T = S_k \cdot (T_k \cdot X \cdot T_k^T) \cdot S_k \quad (4.7)$$

$$X = \hat{C}_k^T \cdot \tilde{Y} \cdot \hat{C}_k = T_k^T \cdot (S_k \cdot \tilde{Y} \cdot S_k) \cdot T_k \quad (4.8)$$

### 4.3 DCT zonale approximative proposée

#### 4.3.1 Transformée proposée

Dans [Potluri 2014], une DCT approximative à 8-points exempte de multiplications et possédant une faible complexité de calcul a été proposée. La DCT approximative améliorée [Potluri 2014] est associée au produit de la matrice entière de faible complexité  $T$  et de la matrice diagonale  $D$  donné ci-dessous :

$$\hat{C} = D \cdot T, \quad (4.9)$$

avec

$$T = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & -1 & -1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 \end{bmatrix} \quad (4.10)$$

et

$$D = \text{Diag} \left( \frac{1}{\sqrt{8}}, \frac{1}{\sqrt{2}}, \frac{1}{2}, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{8}}, \frac{1}{\sqrt{2}}, \frac{1}{2}, \frac{1}{\sqrt{2}} \right) \quad (4.11)$$

Très souvent, la matrice de faible complexité  $T$  a ses éléments définis sur l'ensemble  $\sigma = \{0, \pm 1/2, \pm 1, \pm 2\}$ , et la matrice diagonale  $D$  contient des nombres irrationnels, généralement, sous la forme  $1/\sqrt{n}$  ( $n$  est une petite valeur entière positive) [Lengwehasatit 2004, Bouguezel 2008b, Cintra 2011b].

Les nombres irrationnels dans la matrice diagonale  $D$  impliquent un effort de calcul très élevé, ce qui rend le calcul de la DCT plus complexe et plus coûteux. Cependant, dans le contexte de la compression d'image, la matrice diagonale  $D$  peut simplement être absorbée dans l'étape de quantification dans une chaîne de compression JPEG [Lengwehasatit 2004, Bouguezel 2008b, Cintra 2011b, Tablada 2017]. Par conséquent, dans ce cas, la complexité de l'approximation  $C$  est limitée par la complexité de la matrice de faible complexité  $T$ .

En analysant un ensemble de 50 images de test en niveau de gris de taille  $512 \times 512$  codée à 8-bpp tirées de la base de données d'images [USC-SIPI], nous avons remarqué d'une part, que la DCT approximative améliorée [Potluri 2014] tends à avoir en moyenne 98,07% de son énergie totale d'images concentrée dans les  $4 \times 4$  coefficients de basse fréquence. D'autre part, les coefficients de haute fréquence ont des amplitudes tendent vers zéro. De plus, le processus de quantification dans un codeur de type JPEG vise à réduire à zéro la plupart des coefficients DCT de haute fréquence les moins importants [Rao 2001]. Par conséquent, des économies significatives dans le coût de calcul peuvent être obtenues en ne calculant que les coefficients de basse fréquence les plus importants. À la lumière des considérations ci-dessus, nous proposons la matrice semi-orthogonale  $4 \times 8$  suivante:

$$T_4 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & -1 & -1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}. \quad (4.12)$$

où la DCT zonale approximative correspondante est fournie par :

$$\hat{C}_4 = D_4 \cdot T_4, \quad (4.13)$$

avec  $D_4$  est une matrice diagonale  $4 \times 4$  donnée par :

$$D_4 = \text{Diag} \left( \frac{1}{\sqrt{8}}, \frac{1}{\sqrt{2}}, \frac{1}{2}, \frac{1}{\sqrt{2}} \right). \quad (4.14)$$

Au moyen de la factorisation matricielle [Britanak 2007, Blahut 2010], un algorithme efficace pour le calcul rapide de la matrice  $T_4$  proposée peut être dérivé. En fait,  $T_4$  est une matrice creuse et comporte 16 entrées nulles. Donc, elle peut être décomposée en un produit de quatre matrices creuses sous la forme :

$$T_4 = P \cdot A_3 \cdot A_2 \cdot A_1, \quad (4.15)$$

avec

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$A_3 = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$A_2 = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \text{ et}$$

$$A_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}.$$

où  $P$  est une matrice de permutation qui correspond à la permutation (1324), et  $A_1$ ,  $A_2$  et  $A_3$  sont des matrices de pré-additions.

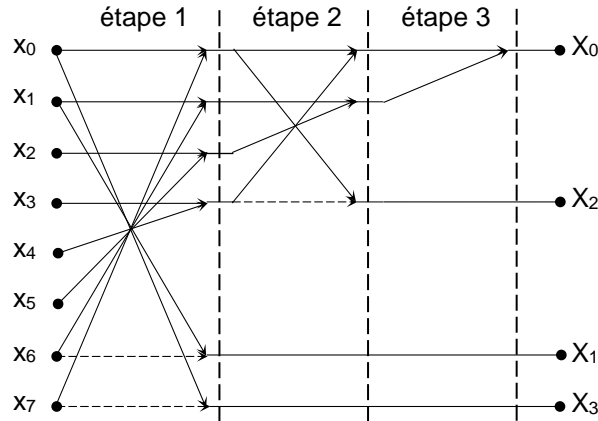
Le graphe de flux correspondant à la factorisation donnée par l'équation (4.15) est illustré sur la figure 4.1. Il est clair d'après le graphe que la transformée proposée nécessite seulement 10 additions, et elle peut être réalisée en trois étapes: la première étape est donnée par la matrice  $A_1$ , la seconde par la matrice  $A_2$  et la dernière étape donnée par la matrice  $A_3$ . Il est à noter également que les étapes numérotées de 1 à 3 doivent être exécutées en série. Cependant, les calculs peuvent être parallélisés à l'intérieur d'une même étape.

En se basant sur l'équation (4.7), la DCT 2-D zonale approximative directe proposée est exprimée comme indiquée ci-dessous et illustrée sur la figure 4.2.

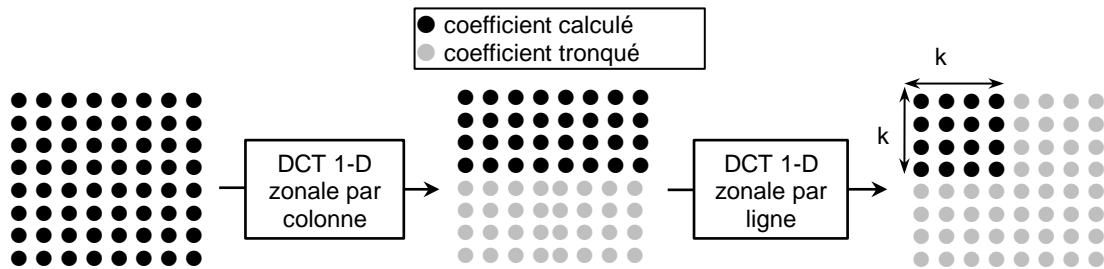
$$\tilde{Y} = \hat{C}_4 \cdot X \cdot \hat{C}_4^T = D_4 \cdot (T_4 \cdot X \cdot T_4^T) \cdot D_4 \quad (4.16)$$

De même, conformément à l'équation (4.8), la DCT 2-D zonale approximative inverse proposée est donnée par :

$$X = \hat{C}_4^T \cdot \tilde{Y} \cdot \hat{C}_4 = T_4^T \cdot (D_4 \cdot \tilde{Y} \cdot D_4) \cdot T_4 \quad (4.17)$$



**Figure 4.1.** Le graphe de flux de la matrice  $T_4$  proposée. Les flèches en pointillés signifient des multiplications par -1.



**Figure 4.2.** DCT 2-D zonale approximative  $4 \times 4$  proposée ( $k = 4$ ).

### 4.3.2 Évaluation de la complexité

Cette section vise à évaluer la complexité de calcul de la DCT 1-D et 2-D zonale approximative proposée. Pour y parvenir, nous avons évalué le coût de calcul de la transformée proposée en comptant le nombre de multiplications, d'additions et d'opérations de décalage de bits et le comparer aux coûts associés aux méthodes de DCT traditionnelles et de l'état de l'art.

Une variété de méthodes de DCT approximatives ont été proposées dans la littérature, parmi lesquelles, nous séparons les approximations significatives suivantes: (1) la DCT approximative introduite par Bouguezel–Ahmad–Swamy (BAS-2013 DCT) [Bouguezel 2013] et sa version zonale [Kouadria 2013]; (2) la DCT arrondie (RDCT) originale [Cintra 2011b] et son approximation zonale [Mechouek 2016]; (3) la DCT arrondie modifié (MRDCT) [Bayer 2012] et la MRDCT zonale [Coutinho 2015]; (4) la DCT améliorée [Potluri 2014]; (5) la transformation proposée dans [Tablada 2017]; (6) l'approximation dans

[Oliveira 2018]; et (7) la DCT approximative dans [Jridi 2017]. Ces approximations ont été choisies parce qu'elles présentent collectivement des complexités arithmétiques extrêmement faibles. De plus, de telles approximations ont démontrées leur utilité dans la compression d'images. Nous avons également évalué la complexité de (i) la DCT exacte calculée selon l'algorithme rapide de Loeffler [Loeffler 1989] et (ii) la DCT zonale exacte [Makkaoui 2010] basée sur le même algorithme de Loeffler.

Soit  $AC_{1-D}(T_k)$  la complexité arithmétique de  $T_k$ . Ainsi, la complexité arithmétique de la DCT 2-D zonale approximative est donnée par :

$$AC_{2-D}(T_k) = 8 \cdot AC_{1-D}(T_k) + k \cdot AC_{1-D}(T_k) \quad (4.18)$$

Pour le cas particulier où  $k$  égale à 4, nous pouvons dériver l'expression ci-dessous :

$$AC_{2-D}(T_4) = 12 \cdot AC_{1-D}(T_4) \quad (4.19)$$

Les complexités arithmétiques obtenues des différentes méthodes de DCT pour le calcul de la DCT 1-D et 2-D sont répertoriés dans le tableau 4.1. Les méthodes de DCT 2-D originales ont été évaluées en utilisant l'équation (4.18), avec  $k = 8$ , tandis que les versions zonales sont évaluées avec l'équation (4.19).

Comme le montre le tableau 4.1, la DCT 2-D zonale approximative proposée possède pratiquement la moitié du coût arithmétique par rapport à la DCT 2-D approximative améliorée originale [Potluri 2014], tout en maintenant un bon compactage d'énergie (98,07% de l'énergie totale moyenne d'images). De plus, nous remarquons que la complexité arithmétique de notre proposition possède le même nombre d'opérations arithmétiques que celle de l'algorithme MRDCT zonale [Coutinho 2015], ayant tous les deux la complexité arithmétique la plus faible parmi tous les algorithmes de DCT approximative significatifs archivés dans la littérature. En comparaison avec d'autres algorithmes de DCT zonale, la méthode proposée révèle des économies significatives en termes du coût arithmétique. Il a été montré respectivement, une réduction de 100% et 58,33% de la complexité multiplicative et additive par rapport à l'algorithme présenté dans [Makkaoui 2010] pour le calcul de la DCT 2-D exacte. En outre, notre proposition a surpassée plusieurs concurrents, notamment l'algorithme BAS-2013 DCT zonale [Kouadria 2013] et RDCT zonale [Mechouek 2016], nécessitant 50% et 37% moins d'opérations pour le calcul de la DCT 1-D et 2-D, respectivement.

Algorithme	DCT 1-D			DCT 2-D		
	Mult.	Add.	Shift	Mult.	Add.	Shift
DCT exacte [Loeffler 1989]	11	29	0	176	464	0
BAS-2013 DCT [Bouguezel 2013]	0	24	0	0	384	0
RDCT [Cintra 2011b]	0	22	0	0	352	0
DCT approximative [Tablada 2017]	0	22	0	0	352	0
DCT approximative [Oliveira 2018]	0	24	6	0	384	96
DCT approximative [Jridi 2017]	0	24	4	0	384	64
MRDCT [Bayer 2012]	0	14	0	0	224	0
DCT améliorée [Potluri 2014]	0	14	0	0	224	0
DCT zonale exacte [Makkaoui 2010]	9	24	0	108	288	0
BAS-2013 DCT zonale [Kouadria 2013]	0	20	0	0	240	0
RDCT zonale [Mechouek 2016]	0	16	0	0	192	0
MRDCT zonale [Coutinho 2015]	0	10	0	0	120	0
DCT proposée	0	<b>10</b>	0	0	<b>120</b>	0

**Table 4.1.** Evaluation de la complexité arithmétique des algorithmes de calcul de la DCT pour un bloc  $8 \times 8$ .

## 4.4 Évaluation des performances

### 4.4.1 Temps d'exécution et consommation d'énergie

Dans cette section, la DCT 2-D zonale approximative proposée est étudiée en termes de temps d'exécution et consommation d'énergie par rapport aux méthodes de l'état de l'art. À cette fin, nous avons effectué des expériences similaires à celles dans [Lee 2009, Lecuire 2012, Kouadria 2013, Mechouek 2016, Araar 2017]. Ainsi, le microcontrôleur ATMEL ATmega128 du nœud-capteur Mica2 et MicaZ [Akyldiz 2002], fonctionnant à 8 MHz avec une consommation électrique active de 22 mW [Polastre 2005] est utilisé comme plate-forme cible.

Le temps d'exécution a été obtenu en utilisant l'outil logiciel WinAVR20100110; un outil de développement open source pour la série d'Atmel AVR des microprocesseurs RISC. La compilation a été effectuée au moyen du simulateur AVR Studio 4 avec l'option

d'optimisation de la vitesse «-O3». La consommation d'énergie est calculée à partir du temps d'exécution  $T$  et la consommation électrique active du microcontrôleur  $P$ , et est donnée par :

$$E = T \times P. \quad (4.20)$$

Le rapport d'amélioration du temps d'exécution – *execution-time improvement ratio* – (ETIR) [Hosny 2008] est également utilisé comme critère d'évaluation pour comparer les différentes méthodes de DCT 2-D. Ce rapport est défini comme suit :

$$ETIR = \left(1 - \frac{Time1}{Time2}\right) \times 100 \quad (4.21)$$

où  $Time1$ , est le temps d'exécution de notre méthode, et  $Time2$  correspond au temps d'exécution de la méthode concurrente. Les résultats sont présentés dans le tableau 4.2.

Algorithme	Temps d'exécution (µs)	ETIR (%)	Energie consommée (µJ)	Taille du code (octets)
DCT exacte [Loeffler 1989]	41 009.88	99.75	902.22	21 300
BAS-2013 DCT [Bouguezal 2013]	192.75	47.40	4.24	2 278
RDCT [Cintra 2011b]	181.50	44.14	3.99	2 102
DCT approximative [Tablada 2017]	181.38	44.11	3.99	2 114
DCT approximative [Oliveira 2018]	205.25	50.61	4.52	2 484
DCT approximative [Jridi 2017]	207.50	51.14	4.57	2 500
MRDCT [Bayer 2012]	161.25	37.13	3.55	1 756
DCT améliorée [Potluri 2014]	147.75	31.38	3.25	1 628
DCT zonale exacte [Makkaoui 2010]	26 196.38	99.61	576.32	13 820
BAS-2013 DCT zonale [Kouadria 2013]	123.25	17.74	2.71	1 370
RDCT zonale [Mechouek 2016]	120.50	15.87	2.65	1 322
MRDCT zonale [Coutinho 2015]	110.38	8.15	2.43	1 142
DCT proposée	101.38	/	2.23	1 032

**Table 4.2.** Temps d'exécution, ETIR, énergie consommée et taille du code d'un bloc  $8 \times 8$  obtenus par le simulateur Atmel AVR studio 4 sur le microcontrôleur Atmel ATmega128.

### **Discussion des résultats**

Il ressort clairement du tableau 4.2 que les deux algorithmes [Loeffler 1989, Makkaoui 2010] consomment plus de temps et d'énergie par rapport aux autres méthodes de DCT. En fait, les deux algorithmes [Loeffler 1989] et [Makkaoui 2010] visent à calculer respectivement, la DCT exacte et la DCT zonale exacte, ce qui nécessite plusieurs opérations de multiplications en virgules flottante. Malgré leur précision, les multiplications à virgule flottante sont extrêmement lentes et consomment beaucoup d'énergie, notamment sur un microcontrôleur contraint en ressources tel que l'ATmega128. Dans ce cas, les algorithmes de la DCT rapide exacte et zonale [Loeffler 1989, Makkaoui 2010] ne sont pas bénéfiques en termes de consommation d'énergie et leur implémentation dans le contexte de réseaux de capteurs n'est pas judicieuse.

Dans l'autre côté, le gain en temps et en énergie pour les méthodes de DCT approximative est très significatif par rapport aux algorithmes [Loeffler 1989, Makkaoui 2010]. Cela peut être justifié par le fait que les méthodes de DCT approximative sont exemptes de multiplications et possèdent des complexités arithmétiques extrêmement faibles.

Comme prévu, les résultats de simulation de l'implémentation logicielle prouvent clairement l'efficacité de la DCT proposée non seulement de point de vue complexité arithmétique mais aussi en termes du temps d'exécution et de consommation d'énergie.

#### **4.4.2 Mémoire requise**

Étant donné que la DCT zonale approximative proposée calcule seulement les 16 coefficients de basse fréquence au lieu de 64 coefficients ; en outre, une table de quantification de  $4 \times 4$  au lieu de  $8 \times 8$  entrées est utilisée, les besoins en mémoire pour stocker les coefficients ainsi que la table de quantification sont considérablement réduits.

La mémoire requise figurant dans la colonne intitulée « *taille du code* » pour la DCT zonale proposé comparée aux autres algorithmes de l'état de l'art est rapportée dans le tableau 4.2. La taille du code indiquée au tableau comprend instructions et données. Il ressort clairement du tableau 4.2 que notre proposition présente la taille du code la plus petite par rapport à toutes les autres méthodes.

### 4.4.3 Compression d'images

Les performances en termes de la qualité d'image de la DCT approximative proposée ont été évaluées en considérant un encodeur de type JPEG et en utilisant les tables de quantification et de codage de Huffman recommandées dans les annexes de la norme JPEG [ITU-T 1992].

Comme évaluation de la qualité d'image, nous avons considéré deux métriques de qualité bien connues et largement utilisées dans la compression d'images, l'indice de similitude structurelle – *Structural SIMilarity* – (SSIM) [Wang 2004] et le rapport signal sur bruit – *Peak Signal to Noise Ratio* – (PSNR) [Rao 2001]. Le SSIM est utilisé pour mesurer la similitude entre l'image reconstruite et l'image originale, tandis que le PSNR représente une mesure de la distorsion induite dans l'image reconstruite. Pour ce faire, un ensemble de 45 images de test en niveau de gris de taille  $512 \times 512$  codées à 8-bpp [USC-SIPI] a été considéré.

Le PSNR est exprimé en décibels (dB), et est défini comme suit:

$$PSNR = 10 \times \log \left( \frac{d^2}{MSE} \right), \quad (4.22)$$

où  $d$  est la valeur de pixel maximale possible de l'image originale (255 dans le cas d'une image en niveau de gris codée sur 8 bits). MSE représente l'erreur quadratique moyenne entre l'image originale  $x_{ij}$  et l'image reconstruite  $y_{ij}$ , et est donnée par l'équation (4.23) suivante :

$$MSE = \frac{1}{N \times M} \sum_{i=1}^M \sum_{j=1}^N (x_{ij} - y_{ij})^2. \quad (4.23)$$

Le SSIM mesure la similitude structurelle entre l'image originale  $x$  et l'image reconstruite  $y$  selon l'expression suivante :

$$SSIM(x, y) = \left( \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \right) \times \left( \frac{2\sigma_{xy} + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \right), \quad (4.24)$$

où  $\mu_x$ ,  $\sigma_x^2$  et  $\sigma_{xy}$  sont respectivement, la moyenne de  $x$ , la variance de  $x$  et la covariance entre  $x$  et  $y$ .  $\mu_y$  et  $\sigma_y^2$  sont la moyenne de  $y$  et la variance de  $y$ , respectivement.  $C_1$  et  $C_2$  sont deux petites constantes (les valeurs par défaut pour  $C_1$  et  $C_2$  sont 0,01 et 0,03) [Wang 2004]. Afin d'obtenir des résultats plus robustes pour notre analyse, toutes les valeurs rapportées sont des

moyennes des mesures calculées pour toutes les images de test considérées. Le tableau 4.3 rapporte les mesures du PSNR et SSIM moyennes données à 0,5 bpp selon la DCT approximative zonale proposée et les méthodes de DCT de l'état de l'art. Les valeurs du PSNR et SSIM de la méthode proposée sont indiquées en gras.

Algorithme	PSNR (dB)	SSIM
DCT exacte [Loeffler 1989]	32.52	0.832
BAS-2013 DCT [Bouguezel 2013]	31.10	0.802
RDCT [Cintra 2011b]	31.35	0.808
DCT approximative [Tablada 2017]	30.02	0.791
DCT approximative [Oliveira 2018]	32.22	0.827
DCT approximative [Jridi 2017]	30.32	0.767
MRDCT [Bayer 2012]	30.02	0.758
DCT améliorée [Potluri 2014]	29.84	0.750
DCT zonale exacte [Makkaoui 2010]	31.65	0.824
BAS-2013 DCT zonale [Kouadria 2013]	30.05	0.794
RDCT zonale [Mechouek 2016]	30.26	0.796
MRDCT zonale [Coutinho 2015]	27.70	0.715
DCT proposée	<b>28.46</b>	<b>0.726</b>

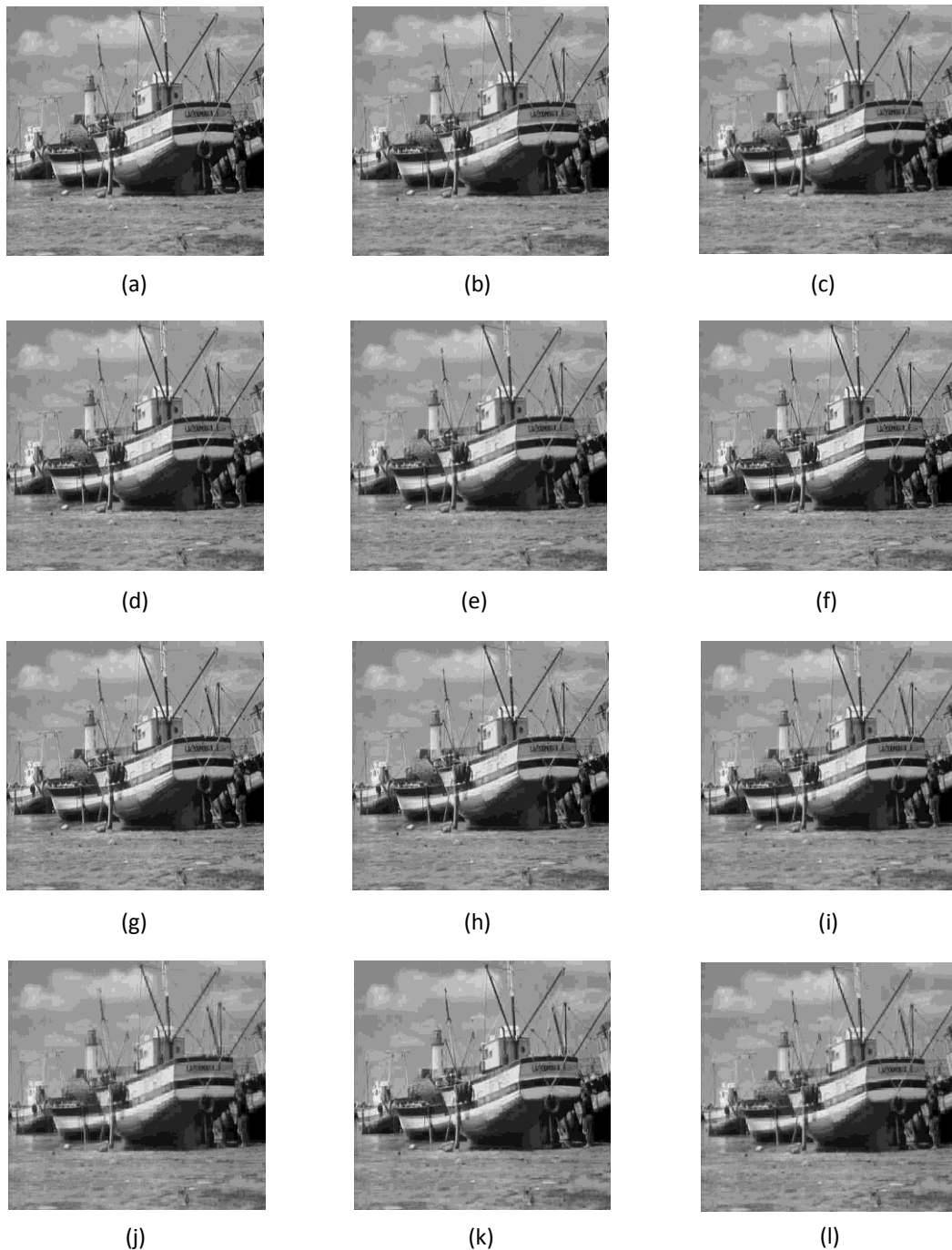
**Table 4.3.** Mesures du PSNR et SSIM moyennes données à 0,5 bpp.

Pour une comparaison qualitative (visuelle), les figures 4.3 et 4.4 montrent respectivement, les images de test Peppers et Boat reconstruites au débit de 0,5 bpp, à partir desquelles nous pouvons clairement voir que la DCT zonale approximative proposée est compétitive par rapport aux autres méthodes.

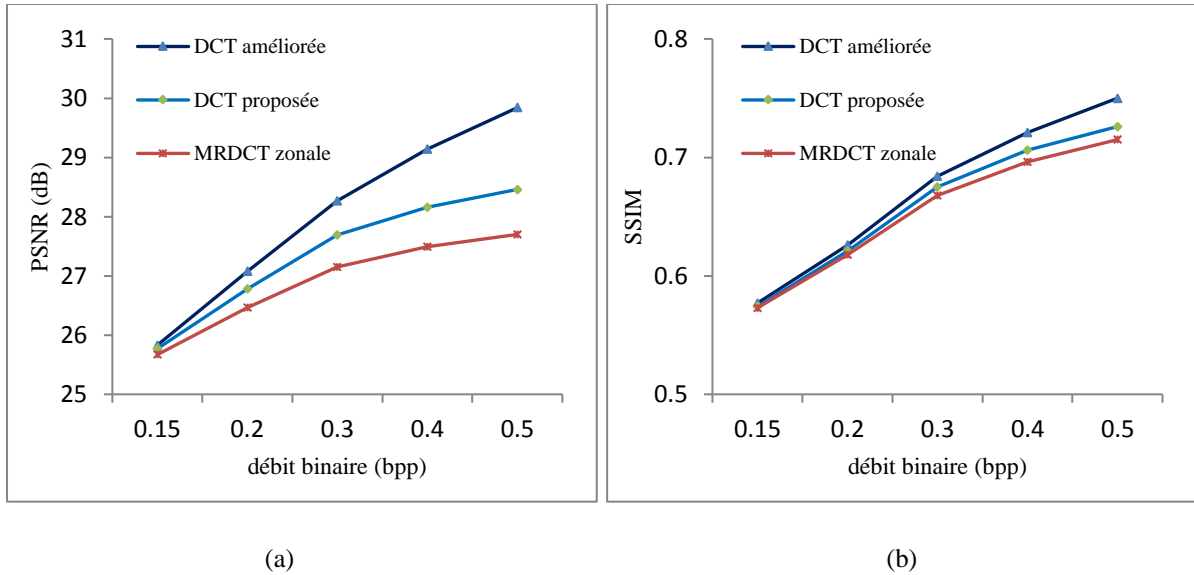
Afin d'étudier davantage l'efficacité de la DCT zonale proposée dans la compression d'images, la figure 4.5 illustre les performances en termes de débit-distorsion données à différents débits binaires (de 0,15 à 0,5 bpp) de la DCT proposée par rapport à la DCT approximative améliorée [Potluri 2014] et à la méthode MRDCT zonale [Coutinho 2015]. Les valeurs moyennes du PSNR et SSIM en fonction du débit binaire sont illustrées sur les figures 4.5-(a) et 4.5-(b), respectivement.



**Figure 4.3.** Image de test Peppers encodée à 0,5 bpp. (a) PSNR=33.96, SSIM=0.866 [Loeffler 1989]. (b) PSNR=31.81, SSIM=0.834 [Bouguezel 2013]. (c) PSNR=29.44, SSIM=0.839 [Kouadria 2013]. (d) PSNR=32.27, SSIM=0.841 [Cintra 2011b]. (e) PSNR=30.03, SSIM=0.831 [Mechouek 2016]. (f) PSNR=30.22, SSIM=0.811 [Tablada 2017]. (g) PSNR=33.54, SSIM=0.858 [Oliveira 2018]. (h) PSNR=30.89, SSIM=0.811 [Jridi 2017]. (i) PSNR=30.89, SSIM=0.807 [Bayer 2012]. (j) PSNR=26.97, SSIM=0.752 [Coutinho 2015]. (k) PSNR=30.61, SSIM=0.804 [Potluri 2014]. (l) PSNR=28.43, SSIM=0.780 [DCT proposée].



**Figure 4.4.** Image de test Boat encodée à 0,5 bpp. (a) PSNR=31.07, SSIM=0.844 [Loeffler 1989]. (b) PSNR=29.39, SSIM=0.809 [Bouguezel 2013]. (c) PSNR=28.03, SSIM=0.798 [Kouadria 2013]. (d) PSNR=29.66, SSIM=0.815 [Cintra 2011b]. (e) PSNR=28.26, SSIM=0.798 [Mechouek 2016]. (f) PSNR=28.27, SSIM=0.792 [Tablada 2017]. (g) PSNR=30.74, SSIM=0.836 [Oliveira 2018]. (h) PSNR=28.71, SSIM=0.783 [Jridi 2017]. (i) PSNR=28.14, SSIM=0.769 [Bayer 2012]. (j) PSNR=25.42, SSIM=0.707 [Coutinho 2015]. (k) PSNR=27.88, SSIM=0.756 [Potluri 2014]. (l) PSNR=26.10, SSIM=0.718 [DCT proposée].



**Figure 4.5.** Performances débit-distorsion. (a) PSNR moyenne. (b) SSIM moyenne.

Les résultats expérimentaux montrent que notre DCT proposée offre un rapport débit-distorsion légèrement inférieur à la transformation DCT améliorée originale [Potluri 2014]. Nous avons observé entre l'intervalle [0,15 - 0,5 bpp], 0,23% - 4,62% de dégradation dans la mesure du PSNR et 0,52% - 3,2% de diminution dans la mesure du SSIM. Malgré ses résultats, notre proposition reste nettement meilleure de point de vue complexité de calcul par rapport à la DCT améliorée [Potluri 2014], pratiquement, c'est la moitié des opérations comme déjà montré dans le tableau 4.1.

Bien que notre proposition et l'algorithme MRDCT zonale [Coutinho 2015] présentent la même complexité arithmétique, il est parfaitement clair de la figure 4.5 que notre proposition surpasse même légèrement le concurrent MRDCT zonale [Coutinho 2015] en termes de performances débit-distorsion.

## 4.5 Architecture VLSI

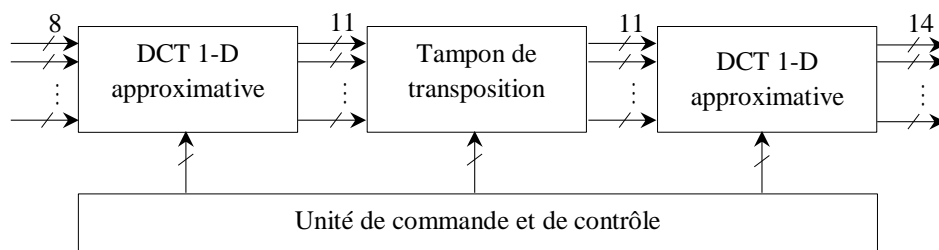
Étant donné que la plupart des plateformes de nœuds-capteurs utilisées dans les réseaux de capteurs sans fil sont très limitées en ressources et manquent de matériel dédié à la compression, l'implémentation logicielle de la DCT 2-D et de la DWT 2-D, est généralement limitée en termes de vitesse de traitement et ne semble pas être une solution très efficace, en particulier pour des applications soumises à des contraintes de temps réel. Par exemple, le temps de traitement de l'algorithme CL-DCT [Lecuire 2012] et de BAS-2013 DCT [Kouadria 2013] pour le calcul de la DCT 2-D sur une image de taille  $512 \times 512$  codée à 8-bpp est

d'environ 2,58 et 1,03 secondes, respectivement. Dans la même veine, le temps de traitement de la transformée DWT approximative à virgule fixe [Phamila 2015] est d'environ 3,95 secondes. En conséquence, les solutions basées sur des implémentations matérielles sont très recherchées afin d'obtenir des performances en temps réel. L'objectif principal derrière, est de proposer une architecture matérielle efficace qui contribue à soulager le microcontrôleur du nœud-captteur du coût de calcul de la DCT, et qui permet de réaliser un traitement ultra-rapide et à faible consommation d'énergie.

Dans cette section, l'architecture matérielle proposée pour le calcul de la transformation DCT 2-D approximative est détaillée. Cette architecture a été initialement décrite en standard VHDL, puis simulée et vérifiée dans le simulateur ModelSim SE PLUS 6.5.

#### 4.5.1 Architecture matérielle proposée

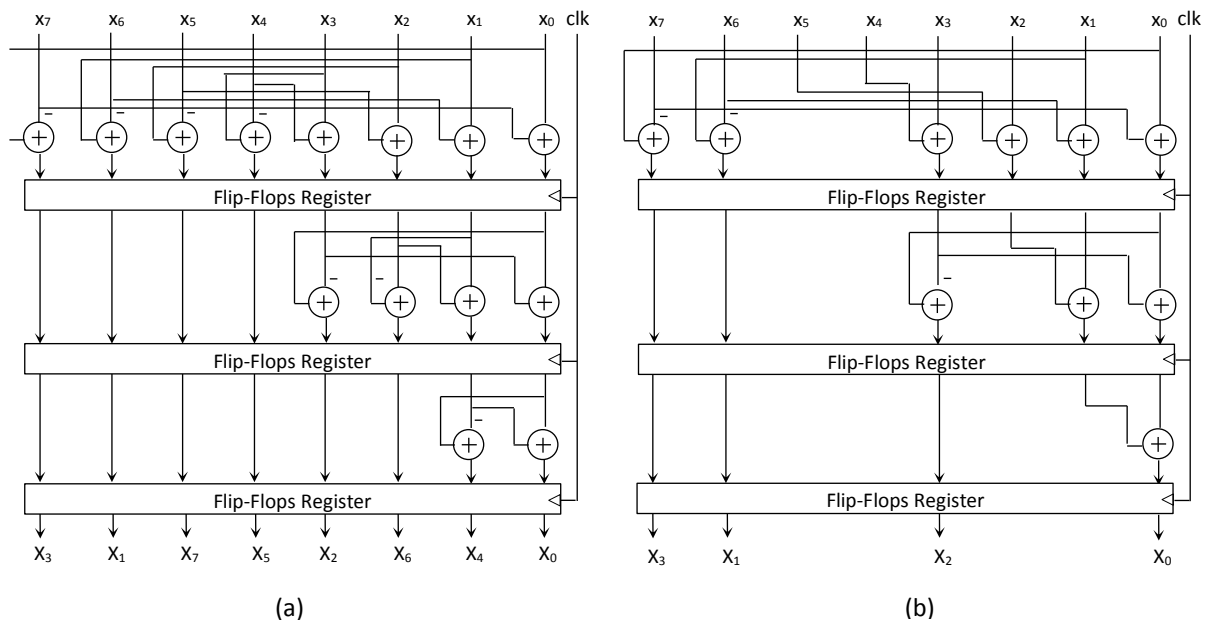
Ense basant sur l'équation (4.7), une architecture parallèle-pipelinée efficace a été conçue pour la DCT 2-D approximative. La figure 4.6 montre un schéma grossier présentant l'architecture matérielle proposée, où une unité de commande et de contrôle basée sur machine à états finis, un tampon de transposition et deux unités de DCT 1-D approximative à 8-points sont impliqués. Les deux unités de DCT 1-D sont fonctionnellement similaires mais les largeurs en bits à l'entrée/sortie de chaque unité sont différentes. La première unité de DCT 1-D effectue une transformée sur les colonnes du bloc d'entrée, et produit les coefficients 1-D codés sur 11 bits, tandis que la seconde fournit une transformation par ligne du résultat intermédiaire, et produit les coefficients 2-D codés sur 14 bits. Quant à l'unité de commande et de contrôle, elle fournit les signaux de commande permettant de gérer le flux de données et d'effectuer la synchronisation des opérations pour les deux unités de DCT 1-D ainsi que le tampon de transposition.



**Figure 4.6.** Architecture parallèle-pipelinée proposée pour le calcul de la DCT 2-D approximative.

#### 4.5.1.1 DCT 1-D approximative

La DCT approximative 1-D est conçue pour être calculée de façon parallèle pipelinée. Le pipeline est fait en trois étapes. Des registres à bascules (Flip-flop) sont insérés le long du chemin de données afin de réduire la longueur du chemin critique, au prix d'un faible coût de la surface totale du circuit. A l'intérieur de chaque étape, les calculs sont parallélisés. De sorte que, plusieurs entrées sont traitées en parallèle et plusieurs sorties sont générées simultanément. La figure 4.7 illustre l'architecture matérielle proposée qui accepte 8 entrées parallèles pour la DCT amélioré [Potluri 2014] et la DCT approximative proposée; et produit respectivement, 8 et 4 sorties parallèles.



**Figure 4.7.** Architecture parallèle-pipelinée proposée pour la DCT 1-D approximative à 8-points. (a) DCT améliorée [Potluri 2014]. (b) DCT proposée.

#### 4.5.1.2 Tampon de transposition

Entre les deux unités de DCT 1-D, un circuit tampon de transposition est requis. Un tel circuit a été conçu pour assurer la transposition du résultat de sortie de la première unité de DCT 1-D, générant le bloc transposé vers l'entrée de la seconde unité de DCT 1-D. Le circuit détaillé du tampon de transposition est représenté sur la figure 4.8-(a), où les connexions  $x_{i,0}, \dots, x_{i,7}$  présentent l'entrée du circuit, et ils sont reliés au dernier étage de la première unité de DCT 1-D pour permettre de récupérer les coefficients 1-D transformés. D'autres part, les broches connecteurs  $X_{0,j}, \dots, X_{7,j}$  présentent la sortie du tampon, et ils sont reliés à

l'entrée de la deuxième unité de DCT 1-D afin de fournir les coefficients 1-D transposés. Quant à l'entrée *index*, elle est reliée à un compteur modulo 8, qui permet de commander simultanément les 8 multiplexeurs placés en parallèle en sélectionnant à chaque fois une ligne différente. Il convient de noter que ce compteur ne sera activé que lorsque tout le tampon est rempli.

Dans le cas de la DCT zonale, le tampon de transposition ressemble au circuit illustré sur la figure 4.8-(b). Il est clair de cette figure qu'il consomme moins de ressources, c'est presque la moitié de ressources.

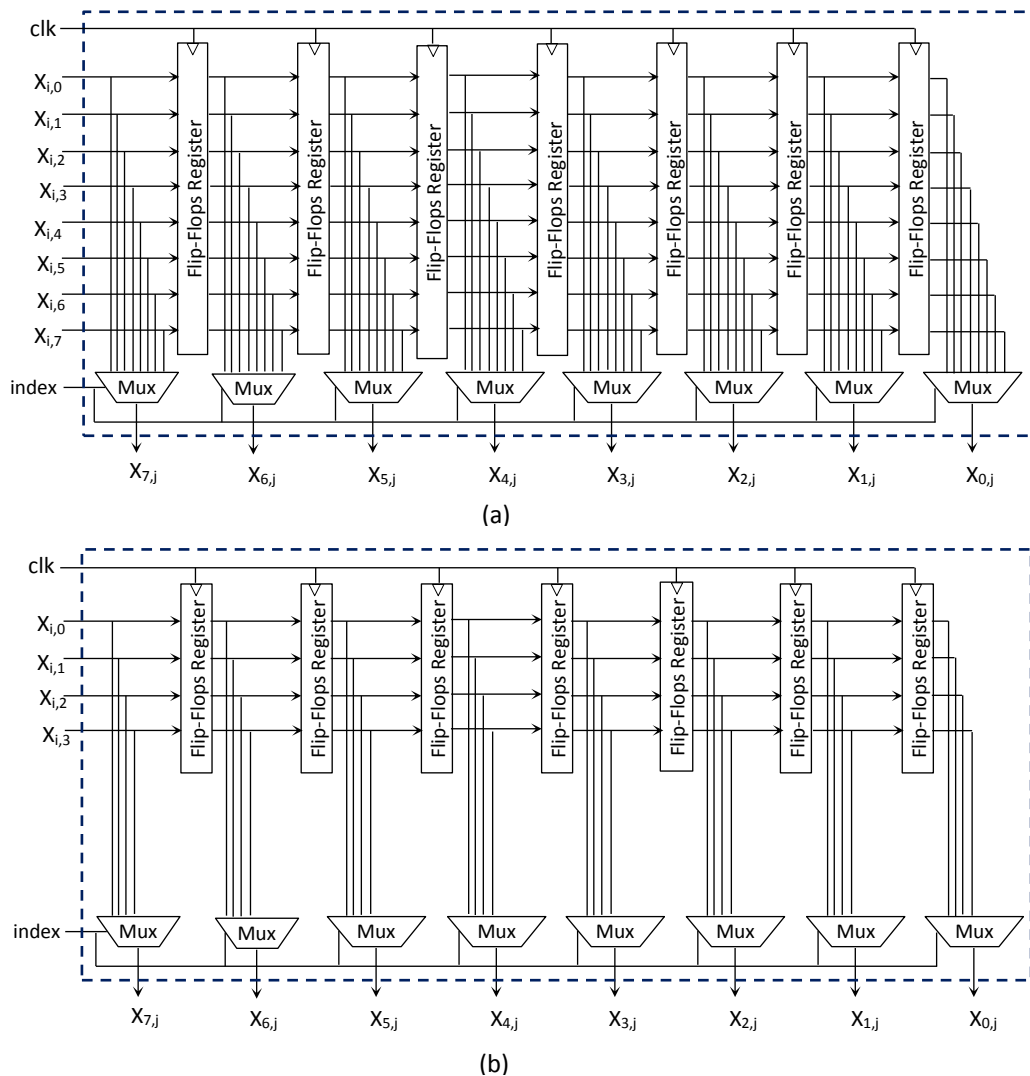


Figure 4.8. Circuit détaillé du tampon de transposition. (a) DCT ordinaire (b) DCT zonale.

## 4.5.2 Implémentation sur FPGA

Nous séparons pour la synthèse et l'implémentation sur FPGA, la DCT approximative proposée, la DCT approximative améliorée [Potluri 2014], l'algorithme MRDCT zonale [Coutinho 2015] et son approximation originale [Bayer 2012]. Tous les algorithmes sélectionnés ont été implémentés en utilisant la même architecture matérielle décrite dans la sous-section 5.1.

La DCT approximative proposée et les méthodes concurrentes ont été synthétisées et implémentées sur un FPGA Xilinx Virtex-6 XC6V5X475T-2ff1156 en utilisant l'outil Xilinx ISE 14.7. Les algorithmes implémentés ont été évalués pour l'utilisation des ressources matérielles, consommation d'énergie et performances temps réel.

Les facteurs de mérite adoptés pour l'évaluation de l'implémentation sur FPGA étaient les suivants: nombre de look-up tables (LUT), nombre de blocs logique configurable occupés (CLB), nombre de registres à bascules flip-flop (FF), période d'horloge minimale ( $T_{cpd}$ ) en nanoseconde (ns) et la fréquence d'horloge maximale ( $F_{max}$ ) en MHz. Les quantités ont été obtenues après placement-et-routage du design à partir du récapitulatif « *Xilinx device utilization summary* » et le rapport « *post-PAR static timing report* ». Il est à noter que les résultats fournis après la synthèse ne sont que des estimations de l'outil ISE. Pour des résultats précis, nous nous référons aux résultats après placement-et-routage. La consommation d'énergie statique ( $Q_p$  en W) et la consommation d'énergie dynamique normalisée ( $D_p$  en mW/MHz) ont été mesurées avec l'outil Xpower Analyzer pour un cycle d'horloge de 100 MHz. Pour une comparaison plus significative, nous avons également rapporté deux métriques largement utilisés dans l'évaluation des circuits VLSI. Il s'agit de la complexité surface-temps – *area-time* – (AT) et la complexité surface-temps-carré – *area-time-square* – (AT<sup>2</sup>); où la surface est mesurée par le nombre de CLB et le temps a été dérivé de  $T_{cpd}$ . La complexité AT est une estimation efficace lorsque la surface de la puce est plus importante que la vitesse. Par contre, la complexité AT<sup>2</sup> est utilisée lorsque la vitesse du circuit est le critère de performance le plus pertinent. Les résultats obtenus pour chaque design sont répertoriés dans le tableau 4.4.

Algorithme	MRDCT [Bayer 2012]	DCT améliorée [Potluri 2014]	MRDCT zonale [Coutinho 2015]	DCT proposée
LUT	643	655	351	372
FF	1058	1058	602	602
CLB	347	335	192	193
$T_{cpd}$ (ns)	2.988	2.899	2.781	2.482
$F_{max}$ (MHz)	334.672	344.947	359.583	402.901
$Q_p$ (W)	5.806	5.806	5.804	5.804
$D_p$ (mW/MHz)	0.55	0.53	0.32	0.32
AT (slices . ns)	1036.84	971.17	533.95	479.03
$AT^2$ (slices . ns <sup>2</sup> )	3098.07	2815.41	1484.92	1188.94

**Table 4.4.** Utilisation des ressources matérielles, vitesse d’horloge et consommation d’énergie de différents designs de DCT 2-D sur un FPGA Xilinx Virtex-6 XC6VSX475T-2ff1156.

Comme le montre le tableau 4.4, la DCT zonale approximative proposée fournit le meilleur  $T_{cpd}$  parmi toutes les méthodes de DCT approximatives conçues. Une telle période d’horloge a permis au design proposé de fonctionner à une fréquence d’horloge supérieure de 10,75%, 14,38% et 16,93% à celle de MRDCT zonale [Coutinho 2015], DCT améliorée [Potluri 2014] et MRDCT [Bayer 2012], respectivement. De plus, la fréquence d’horloge de 402,901 MHz a permis au circuit d’atteindre un débit en temps réel de  $402,901 \times 10^6$  transformations DCT 1-D par seconde, avec un taux de pixels de  $3,223 \times 10^9$  pixels par seconde. Avec ce cycle d’horloge, le design proposé est capable de traiter plus de 12 301 images ( $(512 \times 512)$  pixels, 8 bits / pixel) par seconde. En termes de mesures AT et  $AT^2$ , il apparaît clairement que la DCT approximative proposée surpasse efficacement les méthodes concurrentes. Nous avons également rapporté des économies de 41,82% et 39,62% dans la consommation d’énergie dynamique par rapport à la MRDCT [Bayer 2012] et à la DCT améliorée [Potluri 2014], respectivement.

## 4.6 Conclusion

Dans ce travail, nous avons présenté (i) une DCT zonale approximative de très faible complexité obtenu en appliquant l’approche de la DCT zonale sur la DCT approximative

améliorée [Potluri 2014] (ii) une architecture matérielle parallèle-pipelinée pour le calcul de la DCT zonale approximative proposée et plusieurs autres concurrentes.

La transformée résultante possède une faible complexité de calcul; seulement 10 additions sont nécessaires pour son calcul. Nous avons intégré la DCT approximative proposée dans un encodeur de type JPEG. Les résultats ont montré des performances compétitives dans la compression d'images par rapport aux méthodes de l'état de l'art. Nous avons également conçu une architecture matérielle efficace pour la DCT zonale proposée et nous l'avons implémenté sur un FPGA. Une telle implémentation a révélé des économies considérables de la surface occupée et de la consommation d'énergie avec de meilleures performances en temps réel comparée aux concurrents de l'état de l'art.

À une fréquence d'horloge de 402,901MHz, la conception proposée indiquait un minimum garanti de débit de traitement de données de 25,784 Gbit/s. Ce résultat a montré que l'architecture parallèle-pipelinée proposée pour le calcul de la DCT zonale proposé est l'une des implémentations matérielles les plus rapides. Selon les caractéristiques susmentionnées, la DCT zonale approximative, et l'architecture temps réel et à faible consommation d'énergie proposées sont très appropriées pour l'implémentation dans les réseaux de capteurs sans fil ainsi que dans d'autres systèmes embarqués alimentés par batterie.

# Conclusions et perspectives

## 1. Conclusions

Au cours des dernières décennies, il y a eu une augmentation prolifique de la recherche, du développement et de la commercialisation des réseaux de capteurs sans fil (RCSfs) et de leurs technologies associées. Ce nouveau type de réseau constitué d'un ensemble de nœuds-capteurs, communiquant via une liaison sans fil pour collecter, traiter et transmettre de manière autonome des données dans leur environnement à une station collectrice a pu trouver une pléthore d'applications utiles dans une vaste gamme de domaines, de scénarios et de disciplines différents. Toutefois, les nœuds-capteurs constituant ces réseaux souffrent de leurs capacités de calcul, de stockage et de communication réduites, et surtout de leurs énergies embarquées limitées. La contrainte de l'énergie constitue le point critique dans ces réseaux, et elle a motivée une grande partie des problématiques de recherche dans ce domaine.

Cette thèse a porté sur la compression d'images à la source appliquée aux réseaux de capteurs sans fil en considérant la contrainte de la consommation d'énergie. L'objectif principal était de réduire la complexité algorithmique du standard de compression d'images JPEG. Cet algorithme repose sur une étape de dé-corrélation des données basée sur la transformée DCT, qui est la plus gourmande en temps et en énergie. Nous nous sommes alors intéressés essentiellement à la réduction du coût de calcul de cette étape. Pour y parvenir, nous avons proposé de combiner une méthode de sélection zonale des coefficients avec une méthode de DCT approximative. La combinaison de ces deux méthodes a permis de concevoir une nouvelle DCT zonale approximative présentant une faible complexité de calcul. En intégrant cette DCT dans un encodeur de type JPEG, cette dernière a pu montrer des performances compétitives dans la compression d'images par rapport aux méthodes de l'état de l'art. De plus, les expérimentations de simulation menées sur la DCT proposée ont révélée des économies significatives d'énergie et du temps d'exécution, comme il a été montré par les résultats expérimentaux fournis dans cette thèse.

Nous nous sommes également intéressées à la conception d'une architecture matérielle pour le calcul de la DCT zonale approximative proposée. Cette architecture a été synthétisée et implémentée sur un circuit intégré basé sur la technologie FPGA. Pour montrer l'efficacité de notre implémentation, nous l'avons évaluée pour l'utilisation de ressources matérielles,

consommation d'énergie et performances en temps réel. Les résultats obtenus ont montré clairement la supériorité du design proposé par rapport aux méthodes concurrentes conçues.

## 2. Perspectives

Avec les résultats significatifs et convaincants présentés dans cette thèse, nos travaux de recherches ont fait des progrès importants dans le développement de méthodes de compression d'images efficaces en énergie dans le contexte de réseaux de capteurs sans fil. Toutefois, il existe encore de nombreuses directions de recherche qui méritent d'être explorées pour compléter ces travaux. Nous les structurons comme suit :

- Dans certains scénarios, comme la surveillance de l'habitat, l'utilisateur final ne s'est intéressé que par une partie des informations véhiculées à travers l'image. Dans ce cas, il faut s'intéresser à étudier les régions d'intérêt – *Region of Interest* – (ROI) dans l'image. Il s'agit tout d'abord de détecter et d'extraire le ROI de l'image source. Ensuite, seulement ce ROI sera compressé et transmis via le réseau afin de conserver de l'énergie.
- Comme la compression et la transmission d'images sur les réseaux de capteurs sans fil sont étroitement liées. Il serait mieux de considérer la transmission d'images efficace en énergie en tenant compte de la contrainte des pertes des paquets. Cette dernière contrainte est indispensable car l'image reçue au puits doit avoir une qualité visuelle satisfaisante pour l'utilisateur final. L'objectif est de proposer des solutions qui permettent de fournir un bon compromis entre l'énergie consommée par les nœuds-capteurs et la qualité des images reçues. Le critère de performance peut être exprimé sous la forme d'un rapport énergie-distorsion, incluant la consommation d'énergie de la compression et celle de la transmission.

Sur le plan conception et implémentation matérielle nous prévoyons :

- La prise en compte de plusieurs niveaux de précision de données d'entrée (4-, 8-, 12- et 16-bit) afin d'étudier les performances de l'architecture matérielle conçue à différents niveaux de degrés de précision numérique.
- L'implémentation de l'architecture matérielle conçue pour la DCT zonale approximative proposée sur circuit intégré ASIC technologie CMOS 45 nm.

- La conception et l'implémentation des étapes restantes de l'encodeur JPEG, à savoir l'étape de quantification et l'étape de codage entropique.

## Liste des publications

### Revue internationale avec comité de lecture

- Araar C., Ghanemi S., Benmohammed M., Ateoui H.: *Pruned improved eight-point approximate DCT for image encoding in visual sensor networks requiring only ten additions*. Journal of real-time image processing. pp. 1–12 (2019).

### Conférences internationales avec comité de lecture

- Araar C., Ghanemi S., Benmohammed M., Bourenane E.: *Low complexity image compression using pruned 8-point DCT approximation in wireless visual sensor networks*. In: International Conference on Mathematics and Information Technology (ICMIT), Adrar, Algeria. pp. 279–285 (2017).
- Araar C., Ghanemi S., Benmohammed M., Bourenane E.: *Comparative Study between fast DCT algorithms for image compression in wireless visual sensor networks*. In: 3<sup>rd</sup> International Conference on Networking and Advanced Systems (ICNAS), Annaba, Algeria. pp. 1–4 (2017).

## Références

### A

- [Abadir 2005] Abadir K.M., Magnus J.R.: *Matrix Algebra*. Cambridge University Press, Cambridge (2005)
- [Abe 2007] Abe Y., Iiguni Y.: *Image restoration from a downsampled image by using the DCT*. Signal Processing. **87**(10), pp. 2370–2380 (2007)
- [Ahmed 1974] Ahmed N., Natarajan T., Rao K.R.: *Discrete cosine transform*. Computers. IEEE Transactions on Computers. **C-23**(1), pp. 90-93 (1974)
- [Akkaya 2005] Akkaya K., Younis M.: *A survey on routing protocols for wireless sensor networks*. Ad Hoc Networks. **3**(3), pp. 325–349 (2005)
- [Akyildiz 2002] Akyildiz I.F., Su W., Sankarasubramaniam Y., Cayirci E.: *Wireless sensor networks: a survey*. Computer Networks. **38**(4), pp. 393–422 (2002)
- [Akyildiz 2007] Akyildiz I.F., Melodia T., Chowdhury K.R.: *A survey on wireless multimedia sensor networks*. Computer Networks. **51**(4), pp. 921–960 (2007)
- [Akyildiz 2010] Akyildiz I.F., Vuran M.C.: *Factors Influencing WSN Design*. In: Wireless Sensor Networks, John Wiley & Sons Ltd., West Sussex, UK (2010)
- [Alam 2005] Alam M., Badawy W., Jullien G.: *A new time distributed DCT architecture for MPEG-4 hardware reference model*. IEEE Transactions on Circuits and Systems for Video Technology. **15**(5), pp. 726–730 (2005)
- [Alippi 2007] Alippi C., Anastasi G., Galperti C., Mancini F., Roveri M.: *Adaptive Sampling for Energy Conservation in Wireless Sensor Networks for Snow Monitoring Applications*. In: IEEE International Conference on Mobile Adhoc and Sensor Systems, Pisa, Italy. pp. 1–6 (2007)
- [Al-karaki 2004] Al-karaki J.N., Kamel A.E.: *Routing techniques in wireless sensor networks: a survey*. IEEE Wireless Communications. **11**(6), pp. 6–28 (2004)
- [Alturki 2007] Alturki F.T., Almutairi A.F., Mersereau R.M.: *Analysis of blind data hiding using discrete cosine transform phase modulation*. Signal Processing: Image Communication. **22**(4), pp. 347–362 (2007)
- [Amiya 2010] Amiya N., Ivan S.: *Wireless Sensor and Actuator Networks: Algorithms and Protocols for Scalable Coordination and Data Communication*. Wiley-Interscience New York, NY, USA (2010)
- [Anastasi 2009] Anastasi G., Conti M., Francesco M.D., Passarella A.: *Energy conservation in wireless sensor networks: A survey*. Ad Hoc Networks. **7**(3), pp. 537–568 (2009)
- [Araar 2015] Araar C.: *Compression d'images dans les réseaux de capteurs sans fil*. Mémoire de magister, Université Badji Mokhtar – Annaba (2015)
- [Araar 2017] Araar C., Ghanemi S., Benmohammed M., Bourenane E.: *Low complexity image compression using pruned 8-point DCT approximation in wireless visual sensor networks*. In: International Conference on Mathematics and Information Technology (ICMIT), Adrar, Algeria. pp. 279–285 (2017)
- [Araar 2019] Araar C., Ghanemi S., Benmohammed M., Ateoui H.: *Pruned improved eight-point approximate DCT for image encoding in visual sensor networks requiring only ten additions*. Journal of real-time image processing. pp. 1–12 (2019)
- [Arora 2004] Arora A., Dutta P., Bapat S., Kulathumani V., Zhang H., Naik V., Mittal V., Cao H., Demirbas M., Gouda M., Choi Y., Herman T., Kulkarni S., Arumugam U., Nesterenko M., Vora A.: *A line in the sand: a wireless sensor network for target detection, classification, and tracking*. Computer Networks. **46**(5), pp. 605–634 (2004)

**B**

- [Bayer 2012] Bayer F.M., Cintra R.J.: *DCT-like transform for image compression requires 14 additions only*. Electronic Letters. **48**(15), pp. 919–921 (2012)
- [Benjeddou 2015] Bendjeddou A.: *Prolongation de la Durée de Vie des Batteries dans les Réseaux de Capteurs Sans Fil (RCSF)*. Thèse de doctorat, Université Badji Mokhtar – Annaba (2015)
- [Bhaskaran 1997] Bhaskaran V., Konstantinides K.: *Image and Video Compression Standards*. Kluwer Academic Publishers, Boston (1997)
- [Biagioni 2002] Biagioni E.S., Bridges K.W.: *The application of remote sensor technology to assist the recovery of rare and endangered species*. The international Journal of High Performance Computing Applications. **16**(3), pp. 315–324 (2002)
- [Blahut 2010] Blahut R.E.: *Fast Algorithms for Signal Processing*. Cambridge University Press, Cambridge (2010)
- [Blinn 1993] Blinn J.F.: *What's the deal with the DCT?*. IEEE Computer Graphics and Applications. **13**(4), pp. 78–83 (1993)
- [Bouguezel 2008a] Bouguezel S., Ahmad M.O., Swamy M.N.S.: *A multiplication-free transform for image compression*. In: 2<sup>nd</sup> International Conference on Signals, Circuits and Systems, Monastir, Tunisia. pp. 1–4 (2008)
- [Bouguezel 2008b] Bouguezel S., Ahmad M.O., Swamy M.N.S.: *Low-complexity  $8 \times 8$  transform for image compression*. Electronic Letters. **44**(21), pp. 1249–1250 (2008)
- [Bouguezel 2009] Bouguezel S., Ahmad M.O., Swamy M.N.S.: *A fast  $8 \times 8$  transform for image compression*. In: Proceedings of the International Conference of Microelectronics (ICM), Marrakech, Morocco. pp. 74–77 (2009)
- [Bouguezel 2010] Bouguezel S., Ahmad M.O., Swamy M.N.S.: *A novel transform for image compression*. In: Proc. 53<sup>rd</sup> IEEE International Midwest Symposium on Circuits and Systems (MWSCAS), Seattle, WA, USA. pp. 509–512 (2010)
- [Bouguezel 2011] Bouguezel S., Ahmad M.O., Swamy M.N.S.: *A low-complexity parametric transform for image compression*. In: Proceeding of the IEEE International Symposium on Circuits and Systems (ISCAS), Rio de Janeiro, Brazil. pp. 2145–2148 (2011)
- [Bouguezel 2013] Bouguezel S., Ahmad M.O., Swamy M.N.S.: *Binary discrete cosine and hartley transforms*. IEEE Transaction on Circuits and Systems. **60**(4), pp. 989–1002 (2013)
- [Brahimi 2011] Brahimi N.: *Développement et implémentation des algorithmes de compression d'images basés sur des transformées entières*. Mémoire de magister, Université Ferhat Abbas, Sétif (2011)
- [Britanak 2007] Britanak V., Yip P.C., Rao K.R.: *Discrete Cosine and Sine Transforms: General Properties, Fast Algorithms and Integer Approximations*. Academic Press (2007)

**C**

- [Campbell 2005] Campbell, J., Gibbons, P. B., Nath, S., Pillai, P., Seshan, S. & Sukthankar, R.: *Irisnet: an internet-scale architecture for multimedia sensors*. In: Proceedings of the 13<sup>th</sup> annual ACM international conference on Multimedia (MULTIMEDIA'05), New York, NY, USA. pp. 81–88 (2005)
- [Cardei 2005] Cardei M., Thai M.T., Li Y., Wu W.: *Energy-Efficient Target Coverage in Wireless Sensor Networks*. In: Proceedings IEEE 24<sup>th</sup> Annual Joint Conference of the IEEE Computer and Communications Societies, Miami, Florida, USA. pp. 1976–1984 (2005)

- [Carreto-Castro 1993] Carreto-Castro M.F., Ramirez J.M., Ballesteros J.L., Baez-Lopez D.: *Comparison of lossless compression techniques*. In: Proceedings of 36<sup>th</sup> Midwest Symposium on Circuits and Systems, Detroit, MI, USA. pp. 1268–1270 (1993)
- [Chang 2007] Chang C.-C., Chen S.-G., Chiang J.-C.: *Efficient encoder design for JPEG2000 EBCOT context formation*. In: 15<sup>th</sup> European Signal Processing Conference (EUSIPCO), Poznan, Poland. pp. 644–648 (2007)
- [Chen 1977] Chen W.A., Harrison C., Fralick S.C.: *A Fast computational Algorithm for the Discrete Cosine Transform*. IEEE Transactions on Communications. **25**(9), pp. 1004–1011 (1977)
- [Chen 2008] Chen P., Ahammad P., Boyer C., Huang S.-I., Lin L., Lobaton E., Meingast M., Oh S., Wang S., Yan P., Yang A.Y., Yeo C., Chang L.-C., Tygar J.D., Sastry S.S.: *CITRIC: A Low-Bandwidth Wireless Camera Network Platform*. In: Second ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC), Stanford, CA, USA. pp. 1–10 (2008)
- [Chew 2009] Chew L.W., Chia W.C., Ang L.-M., Seng K.P.: *Very low memory wavelet compression architecture using strip-based processing for implementation in wireless sensor networks*. EURASIP Journal on Embedded System. **2009**, pp. 1–16 (2009)
- [Chiang 2004] Chiang J.-S., Chang C.-H., Y.S. Hsieh Lin, C.-Y., Hsia C.-H.: *High-speed EBCOT with dual context-modeling coding architecture for JPEG2000*. In: IEEE International Symposium on Circuits and Systems, Vancouver, Canada. pp. 865–868 (2004)
- [Chiasserini 2002] Chiasserini C.-F., Magli E.: *Energy consumption and image quality in wireless video-surveillance networks*. In: Proceedings of the 13<sup>th</sup> IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, Lisboa, Portugal. pp. 2357–2361 (2002)
- [Choi 2008] Choi H.J., Seo Y.H., Yoo J.S., Kim D.W.: *Digital watermarking technique for holography interference patterns in a transform domain*. Optics and Lasers in Engineering. **46**(4), pp. 343–348 (2008)
- [Cintra 2011a] Cintra R. J.: *An integer approximation method for discrete sinusoidal transforms*. Journal of Circuits, Systems and Signal processing. **30**(6), pp. 1481–1501 (2011)
- [Cintra 2011b] Cintra R.J., Bayer F.M.: *A DCT approximation for image compression*. IEEE Signal Processing Letters. **18**(10), pp. 579–582 (2011)
- [Clark 1985] Clark R. J.: *Transform Coding of Images*. Academic Press, New York (1985)
- [Coutinho 2015] Coutinho V.A., Cintra R.J., Bayer F.M., Kulasekera S., Madanayake A.: *A multiplierless pruned DCT-like transformation for image and video compression that requires ten additions only*. Journal of Real-time Image Processing. **12**(2), pp. 247–255 (2015)
- [Cristian 2009] Cristian D.-F.: *Transmission d'images sur les réseaux de capteurs sans fil sous la contrainte de l'énergie*. Thèse de doctorat, Université Henri Poincaré, Nancy 1 (2009)
- [Culurciello 2004] Culurciello E., Andree A.G.: *ALOHA CMOS imager*. In: IEEE International Symposium on Circuits and Systems (ISCAS), BC, Vancouver, Canada. pp. 956–959 (2004)
- [Culurciello 2006] Culurciello E., Andreou A.G.: *CMOS image sensors for sensor networks*. Analog Integrated Circuits and Signal Processing. **49**(1), pp. 39–51 (2006)
- [Cărbunar 2006] Cărbunar B., Grama A., Vitek J., Cărbunar O.: *Redundancy and coverage detection in sensor networks*. ACM Transactions on Sensor Networks. **2**(1), pp. 94–128 (2006)

## D

- [Dargie 2010] Dargie W., Poellabauer C.: *Fundamentals of Wireless Sensor Networks Theory and Practice*. Wiley Series on Wireless Communication and Mobile Computing (2010)
- [Dietrich 2009] Dietrich I., Dressler F.: *On the lifetime of wireless sensor networks*. ACM Transactions on Sensor Networks. **5**(1), pp. 1–39 (2009)

[Downes 2006] Downes I., Rad B.L., Aghajan H.: *Development of a Mote for Wireless Image Sensor Networks*. In: Proceeding of Cognitive Systems and Interactive Sensors (COGIS), Paris, France. pp. 1–8 (2006)

[Duhamel 1987] Duhamel P., H'Mida H.: *New 2<sup>n</sup> DCT algorithms suitable for VLSI implementation*. In: IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'87), Dallas, TX, USA. pp. 1805–1808 (1987)

## E

[Enzinger 2012] Enzinger, M.: *Energy-efficient communication in Wireless Sensor Networks*. In: Proceedings of the Seminar Sensor Nodes - Operation, Network and Application. pp. 25–31 (2012)

[Er 2005] Er M.J., Chen W., Wu S.: *High-Speed face recognition based on discrete cosine transform and RBF neural networks*. IEEE Transaction on Neural Networks. **16**(3), pp. 679-691 (2005)

## F

[Feig 1992] Feig E., Winograd S.: *Fast algorithms for the discrete cosine transform*. IEEE Transactions on Signal Processing. **40**(9), pp. 2174–2193, (1992)

[Ferrigno 2005] Ferrigno L., Marano S., Paciello V., Pietrosanto A.: *Balancing computational and transmission power consumption in wireless image sensor networks*. In: IEEE Symposium on Virtual Environments, Human-Computer Interfaces and Measurement Systems (VECIMS), Messina, Italy. pp. 1–6 (2005)

[Friedmann 2007] Friedmann L., Boukhatem L.: *Efficient multi-sink relocation in wireless sensor network*. International Conference on Networking and Services (ICNS'07), Athens, Greece. pp. 90–90 (2007)

[Fugal 2009] Fugal D.L.: *Conceptual Wavelets in Digital Signal Processing: An In-depth, Practical Approach for the Non-mathematician*. Space & Signals Technical Pub. (2009)

## G

[Gall 1992] Gall D.J.L.: *The MPEG video compression algorithm*. Signal Processing: Image Communication. **4**(2), pp. 129–140 (1992)

[Gasparini 2011] Gasparini L., Manduchi R., Gottardi M., Petri D.: *An Ultra-Low Power Wireless Camera Node: Development and Performance Analysis*. IEEE Transactions on Instrumentation and Measurement. **60**(12), pp. 3824–3832 (2011)

[Golomb 1966] Golomb S.W.: *Run-length encodings*. IEEE Transactions on Information Theory. **12**(3), pp. 399–401 (1966)

[Gosselin 2002] Gosselin B.: *La Quantification des Signaux*. Faculté polytechnique de Mons (2002)

[Gresho 1991] Gresho A., Gray R.M.: *Vector quantization and signal Compression*. Kluwer Academic Publishers (1991)

[Guillemot 2004] Guillemot L.: *Une approche vectorielle pour exploiter le contenu de l'image en compression et tatouage*. Thèse de doctorat, Université Henri Poincaré, Nancy 1 (2004)

## H

[Hamdi 2008] Hamdi M., Boudriga N., Obaidat M.S.: *Bandwidth-effective design of a satellite-based hybrid wireless sensor network for mobile target detection and tracking*. IEEE Systems Journal. **2**(1), pp. 74–82 (2008)

[Haweel 2001] Haweel T.I.: *A new square wave transform based on the DCT*. Signal Processing. **82**(11), pp. 2309–2319 (2001)

[Hellman 2006] Hellman K., Colagrosso M.: *Investigating a wireless sensor network optimal lifetime solution for linear topologies*. Journal of Interconnection Networks. **7**(1), pp. 91–99 (2006)

[Hengstler 2007] Hengstler S., Prashanth D., Sufen F., Aghajan, H.: *Mesheye: a Hybrid-Resolution Smart Camera Mote for Applications in Distributed Intelligent Surveillance*. In: 6<sup>th</sup> International Symposium on Information Processing in Sensor Networks, Cambridge, MA, USA. pp. 360–369 (2007)

[Heyne 2006] Heyne B., Sun C.C., Goetze J., Ruan S.J.: *A computationally efficient high-quality cordic based DCT*. In: Proceedings of the 14<sup>th</sup> European Signal Processing Conference (EUSIPCO), Florence, Italy. pp. 1–5 (2006)

[Hosny 2008] Hosny K.M.: *Fast computation of accurate zernike moments*. Journal of Real-Time Image Processing. **3**(1), pp. 97–107 (2008)

[Hou 1987] Hou H.S.: A Fast Recursive Algorithm for Computing the Discrete Cosine Transform. IEEE Transactions on Acoustics, Speech, and Signal Processing. **35**(10), pp. 1455–1461 (1987)

[Howitt 2006] Howitt I., Manges W.W., Kuruganti P., Allgood G., Gutierrez J.A., Conrad J.M.: *Wireless industrial sensor networks: Framework for QoS assessment and QoS management*. ISA Transactions. **45**(3), pp. 347–359 (2006)

[Huffman 1952] Huffman D.A.: *A method for the construction of minimum-redundancy codes*. Proceedings of the IRE. **40**(9), pp. 1098–1101 (1952)

## I

[ITU-T 1990] International Telecommunication Union, ITU-T recommendation H.261 version 1: Video codec for audiovisual services at  $p$  # 64 kbits, ITUT, Tech. Rep. (1990)

[ITU-T 1992] ISO/IEC 10918-1/ITU-T Recommendation T.81, Digital compression and coding of continuous-tone still images. <http://www.jpeg.org/jpeg/>

[ITU-T 1995] International Telecommunication Union, ITU-T recommendation H.263 version 1: Video coding for low bit rate communication, ITU-T, Tech. Rep. (1995)

[ITU-T 2003] Joint Video Team, Recommendation H.264 and ISO/IEC 14 496–10 AVC: Draft ITU-T recommendation and final draft international standard of joint video specification, ITU-T, Tech. Rep. (2003)

[ITU-T 2013] International Telecommunication Union, High efficiency video coding: Recommendation ITU-T H.265, ITU-T Series H: Audiovisual and Multimedia Systems, Tech. Rep. (2013)

## J

[Jadhav 2008] Jadhav D.V., Holambe R.S.: *Radon and discrete cosine transforms based feature extraction and dimensionality reduction approach for face recognition*. Signal Processing. **88**(10), pp. 2604–26 (2008)

[Jeong 1998] Jeong Y., Lee I., Kim H.S., Park K.T.: *Fast DCT algorithm with fewer multiplication stages*. Electronics Letters. **34**(8), pp. 723–724 (1998)

[Jridi 2017] Jridi M., Meher P.K.: *Scalable approximate DCT architectures for efficient HEVC-compliant video coding*. IEEE Transactions on Circuits and Systems for Video Technologie. **27**(8), pp. 1815–1825 (2017)

## K

[Kacimi 2009] Kacimi R.: *Techniques de conservation d'énergie pour les réseaux de capteurs sans fil*. Thèse de doctorat, Université de Toulouse (2009)

[Kaddachi 2012] Kaddachi M.L., Soudani A., Lecuire V., Torki K., Makkaoui L., Moureaux J.-M.: *Low power hardware-based image compression solution for wireless camera sensor networks*. Computer Standards & Interfaces. **34**(1), pp. 14–23 (2012)

[Karl 2005] Karl H., Willig A.: *Protocols and architectures for wireless sensor networks*. John Wiley & Sons Ltd., USA (2005)

[Keshab 1999] Keshab K.P., Takao N.: *Digital Signal Processing for Multimedia Systems*. CRC Press (1999)

[Khayam 2003] Khayam S. A.: *The Discrete Cosine Transform (DCT): Theory and Application*. Department of Electrical & Computer Engineering, Michigan State University (2003)

[Kitajima 1976] Kitajima H.: *Energy packing efficiency of the Hadamard transform*. IEEE Transactions on Communications. **24**, pp. 1256–1258 (1976)

[Kouadria 2013] Kouadria N., Doghmane N., Messadeg D., Harize S.: *Low complexity DCT for image compression in wireless visual sensor networks*. Electronic Letters. **49**(24), pp. 1531–1532 (2013)

[Kouadria 2014] Kouadria N. : *Codage et compression d'images pour des réseaux de capteurs sans fil*. Thèse de doctorat, Université Badji-Mokhtar, Annaba (2014)

[Krishnamachari 2002] Krishnamachari L., Estrin D., Wicker S.: *The impact of data aggregation in wireless sensor networks*. In: Proceedings 22<sup>nd</sup> International Conference on Distributed Computing Systems Workshops (ICDCSW), Vienna, Austria. pp. 575–578 (2002)

[Krupinski 2007] Krupinski R., Purczynski J.: *Modeling the distribution of DCT coefficients for JPEG reconstruction*. Signal Processing: Image Communication. **22**(5), pp. 439–447 (2007)

[Kuo 2011] Kuo W.-K., Wu K.-W.: *Traffic prediction and QoS transmission of real-time live VBR videos in WLANs*. ACM Transactions on Multimedia Computing, Communications, and Applications. **7**(4), pp. 1–21 (2011)

## L

[Labrador 2009] Labrador M.A., Wightman P.M.: *Topology Control in Wireless Sensor Networks: with a companion simulation tool for teaching and research*. Springer Netherlands (2007)

[Lecuire 2012] Lecuire V., Makkaoui L., Moureaux J.-M.: *Fast zonal DCT for energy conservation in wireless image sensor networks*. Electronics Letters. **48**(2), pp. 125–127 (2012)

[Lee 1984] Lee B.G.: *A New Algorithm to Compute the Discrete Cosine Transform*. IEEE Transactions on Acoustics, Speech, and Signal Processing. **32**(6), pp. 1243–1245 (1984)

[Lee 2009] Lee D., Kim H., Rahimi M., Estrin D., Villasenor J.D.: *Energy-efficient image compression for resource constrained platforms*. IEEE Transaction on Image Processing. **18**(9), pp. 2100–2113 (2009)

[Lengwehasatit 2004] Lengwehasatit K., Ortega A.: *Scalable variable complexity approximate forward DCT*. IEEE Transactions on Circuits and Systems for Video Technology. **14**(11), pp. 1236–1248 (2004)

[Lian 2003] Lian C.-J., Chen K.-F., Chen H.-H., Chen L.-G.: *Analysis and architecture design of block-coding engine for EBCOT in JPEG2000*. IEEE Transaction on Circuits and Systems for Video Technology. **13**(3), pp. 219–230, (2003)

[Liang 2001] Liang J., Tran T.D.: *Fast multiplierless approximations of the DCT with the lifting scheme*. IEEE Transactions on Signal Processing. **49**(12), pp. 3032–3044 (2001)

[Ling 2012] Ling G., Yifeng H., Sun-Yuan K.: *Multimedia Image and Video Processing*. CRC Press, Boca Raton (2012)

[Liu 2008] Liu Z., Liu C.: *Fusion of the complementary Discrete Cosine Features in the YIQ color space for face recognition*. Computer Vision and Image Understanding. **111**(3), pp. 249–262 (2008)

[Loeffler 1989] Loeffler C., Ligtenberg A., Moschytz G.S.: *Practical fast 1-D DCT algorithms with 11 multiplications*. In: International Conference on Acoustics, Speech, and Signal Processing, Glasgow, UK. pp. 988–991 (1989)

[Lu 1997] Lu N., Lu W.-K. : *Fractal imaging*. Academic Press (1997)

## M

- [Madanayake 2015] Madanayake A., Cintra R.J., Dimitrov V., Bayer F. M., Wahid K.A., Kulasekera S., Edirisuriya A., Potluri U.S., Madishetty S.K., Rajapaksha N.: *Low-Power VLSI Architectures for DCT/DWT: Precision vs Approximation for HD Video Biomedical and Smart Antenna Applications*. IEEE Circuits and Systems Magazine. **15**(1), pp. 25–47 (2015)
- [Mahfoudh 2008] Mahfoudh S., Minet P.: *Survey of energy efficient strategies in wireless ad hoc and sensor networks*. 7<sup>th</sup> International Conference on Networking (ICN), Cancun, Mexico. pp. 1–7 (2008).
- [Makkaoui 2010] Makkaoui L., Lecuire V., Moureaux J.-M.: *Fast zonal DCT-based image compression for wireless camera sensor networks*. In: Proceedings of the IEEE International Conference on Image Processing Theory, Tools and Applications (IPTA'10), Paris, France. pp. 126–129 (2010)
- [Makkaoui 2012] Makkaoui L.: *Compression d'images dans les réseaux de capteurs sans fil*. Thèse de doctorat, Université de Lorraine (2012).
- [Mallat 1989] Mallat S. G.: *A theory for multiresolution signal decomposition: the wavelet representation*. In: IEEE Transactions on Pattern Analysis and Machine Intelligence. pp. 674–693 (1989).
- [Mammeri 2008a] Mammeri A., Khoumsi A., Ziou D., Hadjou B.: *Energy-aware JPEG for visual sensor networks*. In: Maghrebian Conference on Information Technologies (MCSEAI'08), Oran, Algeria. (2008)
- [Mammeri 2008b] Mammeri A., Khoumsi A., Ziou D., Hadjou B.: *Modeling and adapting JPEG to the energy requirements of VSN*. In: Proceedings of the IEEE International Conference on Computer Communications and Networks (ICCCN' 08), US Virgin Islands, USA. pp. 1–6 (2008)
- [Mammeri 2012] Mammeri A., Hadjou B., Khoumsi A.: *A survey of image compression algorithms for visual sensor networks*. ISRN Sensor Networks. **2012**, pp. 1–19 (2012).
- [Markel 1971] Markel J.: *FFT pruning*. IEEE Transactions on Audio and Electroacoustics. **9**(4), pp. 305–311 (1971)
- [Mechouek 2016] Mechouek K., Kouadria N., Doghmane N., Kaddeche N.: *Low complexity DCT approximation for image compression in wireless image sensor networks*. Journal Circuits System and Computer. **25**(8), pp. 1650088 (2016).
- [Mo 2005] Mo W., Qiao D., Wang Z.: *Mostly-Sleeping Wireless Sensor Networks: Connectivity, k-Coverage, and alpha-Lifetime*. In: The 43<sup>rd</sup> Annual Allerton Conference on Communication, Control, and Computing, Monticello, France. (2005)

## O

- [Oliveira 2018] Oliveira, R.S., Cintra, R.J., Bayer, F.M., da Silveira, T.L.T., Madanayake, A., Leite, A.: *Low-complexity 8-point DCT approximation based on angle similarity for image and video coding*. Multidimensional Systems and Signal Processing. pp. 1–32 (2018)

## P

- [Peak 2014] Jeongyeup P., John H., Sharon C., Ramesh G.: *Image-Based Environmental Monitoring Sensor Application Using an Embedded Wireless Sensor Network*. Sensors (Basel). **14**(9), pp. 15981–16002 (2014)
- [Pennebaker 1993] Pennebaker W.B., Mitchell J.L.: *JPEG – Still Image Data Compression Standard*. International Thomson Publishing Inc., New York (1993)
- [Phamila 2015] Phamila A.V.Y., Amutha R.: *Energy-efficient low bit rate image compression in wavelet domain for wireless image sensor networks*. Electronic Letters. **51**(11), pp. 824–826 (2015)
- [Polastre 2005] Polastre J., Szewczyk R., Culler D.: *Telos: Enabling ultra-low power wireless research*. In: 4<sup>th</sup> International Symposium on Information Processing in Sensor Networks, Boise, USA. pp. 364–369 (2005)

[Potluri 2012] Potluri U.S., Madanayake A., Cintra R.J., Bayer F.M., and Rajapaksha N.: Multiplier-free DCT approximations for RF multi-beam digital aperture-array space imaging and directional sensing. *Measurement Science and Technology*. **23**(11), pp. 114003 (2012)

[Potluri 2014] Potluri U.S., Madanayake A., Cintra R.J., Bayer F.M., Kulasekera S., Edirisuriya A.: *Improved 8-Point Approximate DCT for Image and Video Compression Requiring Only 14 Additions*. In: *IEEE Transactions on Circuits and Systems*. **61**(6), pp. 1727-1740 (2014)

[Pottie 2000] Pottie G.J., Kaiser W.J.: *Wireless integrated network sensors*. *Communications of the ACM*. **43**(5), pp. 51–58 (2000)

[Pourazad 2012] Pourazad M.T., Doutre C., Azimi M., Nasiopoulos P.: *HEVC: The new gold standard for video compression: How does HEVC compare with H.264/AVC?*. *IEEE Consumer Electronics Magazine*. **1**(3), pp. 36–46 (2012)

## Q

[Qi 2008] Qi H., Zheng D., Zhao J.: *Human visual system based adaptive digital image watermarking*. *Signal Processing*. **88**(1), pp. 174–178 (2008)

## R

[Rabaey 2000] Rabaey J., Ammer J., da Silva J.L., Patel D.: *Pico-Radio: ad-hoc wireless networking of ubiquitous low-energy sensor/monitor nodes*. In: *Proceedings IEEE Computer Society Workshop on VLSI 2000. System Design for a System-on-Chip Era*, Orlanda, Florida, USA. pp. 9–12 (2000)

[Radha 2003] Radha H.: *Lecture Notes: ECE 802 - Information Theory and Coding*. (2003)

[Raghunathan 2002] Raghunathan V., Schurgers C., Park S., Srivastava M.B.: *Energy-aware wireless microsensor networks*. *IEEE Signal Processing Magazine*. **19**(2), pp. 40–50 (2002)

[Rahimi 2005] Rahimi M., Baer R., Iroezzi O.I., Garcia J.C., Warrior J., Estrin D., Srivastava, M.: *Cyclops: in situ image sensing and interpretation in wireless sensor networks*. In: *Proceedings of the 3<sup>rd</sup> International Conference on Embedded Networked Sensor Systems (SenSys'05)*, USA. pp. 192–204 (2005)

[Rao 1990] Rao K.R., Yip P.: *Discrete Cosine Transform: Algorithms, Advantages, Applications*. Academic Press, Boston (1990)

[Rao 2001] Rao K.R., Yip P.C.: *The Transform and Data Compression Handbook*. CRC Press LLC, Boca Raton (2001)

[Rezaei 2012] Rezaei Z., Mobininejad S.: *Energy Saving in Wireless Sensor Networks*. *International Journal of Computer Science and Engineering Survey*. **3**(1), pp. 1–15 (2012)

[Rissanen 1979] Rissanen G.G.L.: *Arithmetic Coding*. *IBM Journal of Research and Development*. **23**(2), pp. 149-162 (1979)

[Roma 2011] Roma N., Sousa L.: *A tutorial overview on the properties of the discrete cosine transform for encoded image and video processing*. *Signal Processing*. **91**(11), pp. 2443–2464 (2011)

[Rowe 2007] Rowe A., Goode A., Goel D., Nourbakhsh I.: *CMUcam3: An open programmable embedded vision sensor*. Technical Report RI-TR-07-13. Carnegie Mellon Robotics Institute. Pittsburgh, Pennsylvania 15213 (2007)

[Réf. 01] *module 3 Lossy Image Compression Systems: Lesson 9 Discrete Cosine Transforms*. Version 2 ECE IIT, Kharagpur

## S

[Said 1996] Said A., Pearlman W.A.: *A new fast and efficient image codec based on set partitioning in hierarchical trees*. *IEEE Transactions on Circuits and Systems for Video Technology*. **6**(3), pp. 243–250 (1996)

- [Sanderson 2003] Sanderson C., Paliwal K.K.: *Features for robust face-based identity verification*. Signal Processing. **83**(5), pp. 931–940 (2003)
- [Saponara 2012] S. Saponara. Real-time and low-power processing of 3D direct/inverse discrete cosine transform for low-complexity video codec. Journal of Real-Time Image Processing. 7, pp. 43–53 (2012)
- [Seber 2008] Seber G.A.F.: *A Matrix Handbook for Statisticians*. John Wiley & Sons, Inc. (2008)
- [See 2007] See K.W., Loke K.S., Lee P.A., Loe K.F.: *Image reconstruction using various discrete orthogonal polynomials in comparison with DCT*. Applied Mathematics and Computation. **193**(2), pp. 346–359 (2007)
- [Sejdic 2018] Sejdic E., Falk T.H.: *Signal Processing and Machine Learning for Biomedical Big Data*. CRC press (2018)
- [Shapiro 1993] Shapiro J.M.: *Embedded image coding using zerotrees of wavelet coefficients*. IEEE Transaction on Signal Processing. **41**(12), pp. 3445–3462 (1993)
- [Shukla 2010] Shukla J., Alwani M., Tiwari A.K.: *A survey on lossless image compression methods*. In: 2<sup>nd</sup> International Conference on Computer Engineering and Technology (IC CET), Chengdu. pp. 136–141 (2010)
- [Silva 2008] Silva L.C.D.: *Audiovisual sensing of human movements for home-care and security in a smart environment*. International Journal on Smart Sensing and Intelligent Systems. 1(1), pp. 220–245 (2008)
- [Slawecki 1992] Slawecki D., Li W.: *DCT/IDCT processor design for high data rate image coding*. IEEE Transactions on Circuits and Systems for Video Technology. **2**(2), pp. 135–146 (1992)
- [Soro 2009] Soro S., Heinzelman W.: *A survey of visual sensor networks*. Advances in Multimedia. **2009**, pp. 1–19 (2009)
- [Strang 1999] Strang G.: *The Discrete Cosine Transform*. Society for Industrial and Applied Mathematics. **41**(1), pp. 135–147 (1999)
- [Suehiro 1986] Suehiro N., Hatori M.: *Fast Algorithms for the DFT and other Sinusoidal Transforms*. IEEE Transactions on Acoustics, Speech, and Signal Processing. **34**(3), pp. 642–644 (1986)
- [Suseela 2018] Suseela G., Asnath V.P.Y.: *Energy efficient image coding techniques for low power sensor nodes: A review*. Ain Shams Engineering Journal. **9**(4), 2961–2972 (2018)

## T

- [Tablada 2017] Tablada C.J., da Silveira T.L.T., Cintra R.J., Bayer F.M.: *DCT approximations based on Chen's factorization*. Signal Processing: Image Communication. **58**, pp. 14–23 (2017)
- [Taubman 2000] Taubman D.: *High performance scalable image compression with EBCOT*. IEEE Transactions on Image Processing. **9**(7), pp. 1158–1170 (2000)
- [Taylor 2002] Taylor C.N., Panigrahi D., Dey S.: *Design of an adaptive architecture for energy efficient wireless image communication*. Lecture Notes in Computer Science. **2268**, 260–273 (2002)
- [Teixeira 2006] Teixeira T., Lymberopoulos D., Culurciello E., Aloimonos Y., Savvides, A.: *A Lightweight Camera Sensor Network Operating on Symbolic Information*. In: Proceedings of the 1<sup>st</sup> Workshop on Distributed Smart Cameras. (2006)
- [Tian 2002] Tian D., Georganas N.D.: *A coverage-preserving node scheduling scheme for large wireless sensor networks*. In: Proceedings of the 1<sup>st</sup> ACM international workshop on Wireless sensor networks and applications (WSNA'02), Atlanta, Georgia, USA. pp. 32–41 (2002)
- [Tilak 2002] Tilak S., Abu-Ghazaleh N.B., Heinzelman W.: *A taxonomy of wireless micro-sensor network models*. ACM SIGMOBILE Mobile Computing and Communications Review. **6**(2), pp. 28–36 (2002)

## U

[USC-SIPI] The USC-SIPI image database. University of Southern California, Signal and Image Processing Institute. <http://sipi.usc.edu/database/>

## V

[Vetterli 1984] Vetterli M., Nussbaumer H.: *Simple FFT and DCT Algorithms with Reduced Number of Operations*. Signal Processing. **6**(4), pp. 267–278 (1984)

## W

[Wallace 1992] Wallace G.K.: *The JPEG still picture compression standard*. IEEE Transactions on Consumer Electronics. **38**(1), pp. xviii-xxxiv (1992)

[Wang 1984] Wang Z.: *Fast Algorithms for the Discrete W-Transform and for the Discrete Fourier Transform*. IEEE Transactions on Acoustics, Speech, and Signal Processing. **32**(4), pp. 803-816 (1984)

[Wang 1991] Wang Z.: *Pruning the fast discrete cosine transform*. IEEE Transaction on Communications. **39**(5), pp. 640–643 (1991)

[Wang 2004] Wang Z., Bovik A.C., Sheikh H.R., Simoncelli E.P.: *Image quality assessment: from error visibility to structural similarity*. IEEE Transaction on Image Processing. **13**, pp. 600–612 (2004)

[Wang 2010] Wang Q., Balasingham I.: *Wireless Sensor Networks - An Introduction, Wireless Sensor Networks: Application - Centric Design*. Geoff V Merrett and Yen Kheng Tan, IntechOpen (2010)

[Wawerla 2008] Wawerla J., Marshall S., Mori G., Rothley K., Sabzmejdani P.: *Bearcat: Automated wildlife monitoring at the Arctic Circle*. Machine Vision and Applications. **20**(5), pp. 303–317 (2009)

[White 1989] White S.A.: *Applications of distributed arithmetic to digital signal processing: a tutorial review*. IEEE ASSP Magazine. **6**(3), pp. 4–19 (1989)

[Wiegand 2003] Wiegand T., Sullivan G.J., Bjontegaard G., Luthra A.: *Overview of the H.264/AVC video coding standard*. IEEE Transactions on Circuits and Systems for Video Technology. **13**(7), pp. 560–576, (2003)

[Wong 2007] Wong K., Qi X., Tanaka K.: *A DCT-based Mod4 steganographic method*. Signal Processing. **87**(6), pp. 1251-1263 (2007)

[Wu 2003] Wu M., Chen C.W.: *Multiple bitstream image transmission over wireless sensor networks*. In: Proceedings of IEEE Sensors, Toronto, Canada. pp. 727–731 (2003)

[Wu 2004] Wu H., Abouzeid A.A.: *Power aware image transmission in energy constrained wireless networks*. In: Proceedings of The 9<sup>th</sup> IEEE Symposium on Computers and Communications (ISCC), Alexandria, Egypt. pp. 202–207 (2004)

[Wu 2005] Wu K., Gao Y., Li F., Xiao Y.: *Lightweight Deployment-Aware Scheduling for Wireless Sensor Networks*. Mobile Networks and Applications. **10**(6), pp. 837–852 (2005)

## Y

[Yang 2005] Yang M., Bourbakis N.: *An overview of lossless digital image compression techniques*. In: Proceedings of 48<sup>th</sup> Midwest Symposium on Circuits and Systems, Covington, KY, USA. pp. 1099-1102 (2005)

[Yu 2002] Yu S., Swartzlander E.E.: *A scaled DCT architecture with the CORDIC algorithm*. IEEE Transactions on Signal Processing. **50**(1), pp. 160–167 (2002)

[Yu 2004] Yu W., Sahinoglu Z., Vetro A.: *Energy efficient JPEG2000 image transmission over wireless sensor networks*. In: IEEE Global Telecommunications Conference (GLOBECOM'04), Dallas, USA. pp. 2738–2743 (2004)

[Yuan 2006] Yuan W., Hao P., Xu C.: *Matrix Factorization for Fast DCT Algorithms*. In: IEEE International Conference on Acoustics Speech and Signal Processing Proceedings (ICASSP), Toulouse, France. pp. 948-951 (2006)

**Z**

[ZainEldin 2015] ZainEldin H., Elhosseini M.A, Ali H.A.: *Image compression algorithms in wireless multimedia sensor networks: A survey*. Ain Shams Engineering Journal. **6**(2), pp. 481-490 (2015)

[Ze-Nian 2014] Ze-Nian L., Mark S.D., Jiangchuan L.: *Fundamentals of Multimedia*. Springer, Cham (2014)