



**HAL**  
open science

# Compression d'images dans les réseaux de capteurs sans fil

Leila Makkaoui

► **To cite this version:**

Leila Makkaoui. Compression d'images dans les réseaux de capteurs sans fil. Autre. Université de Lorraine, 2012. Français. NNT : 2012LORR0416 . tel-01750619v1

**HAL Id: tel-01750619**

**<https://hal.univ-lorraine.fr/tel-01750619v1>**

Submitted on 29 Mar 2018 (v1), last revised 28 Feb 2013 (v2)

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



## AVERTISSEMENT

Ce document est le fruit d'un long travail approuvé par le jury de soutenance et mis à disposition de l'ensemble de la communauté universitaire élargie.

Il est soumis à la propriété intellectuelle de l'auteur. Ceci implique une obligation de citation et de référencement lors de l'utilisation de ce document.

D'autre part, toute contrefaçon, plagiat, reproduction illicite encourt une poursuite pénale.

Contact : [ddoc-theses-contact@univ-lorraine.fr](mailto:ddoc-theses-contact@univ-lorraine.fr)

## LIENS

Code de la Propriété Intellectuelle. articles L 122. 4

Code de la Propriété Intellectuelle. articles L 335.2- L 335.10

[http://www.cfcopies.com/V2/leg/leg\\_droi.php](http://www.cfcopies.com/V2/leg/leg_droi.php)

<http://www.culture.gouv.fr/culture/infos-pratiques/droits/protection.htm>



UNIVERSITÉ  
DE LORRAINE

FACULTE DES SCIENCES & TECHNOLOGIES

CENTRE DE RECHERCHE  
EN AUTOMATIQUE DE  
NANCY



CNRS UMR 7039

U.F.R. : Sciences et Techniques Mathématiques, Informatique et Automatique

Ecole Doctorale : IAEM Lorraine

Département de Formation Doctorale : Automatique

Thèse  
présentée pour l'obtention du titre de  
Docteur de l'Université de Lorraine  
en Sciences, spécialité Automatique,  
Traitement du Signal et Génie Informatique  
par **Makkaoui** LEILA

# Compression d'images dans les réseaux de capteurs sans fil

Soutenue publiquement le 26 Novembre 2012

Membres du jury :

<i>Rapporteurs :</i>	M. Jean-Jacques Pansiot	Professeur, Université de Strasbourg
	M. Amine Nait-Ali	Professeur, Université Paris-Est Créteil
<i>Examineurs :</i>	M. Francis Lepage	Professeur, Université de Lorraine
	M. Congduc Pham	Professeur, Université de Pau
	M. Jean-Marie Moureaux	Professeur, Université de Lorraine (Directeur de thèse)
	M. Vincent Lecuire	MdC, Université de Lorraine (Co-Directeur de thèse)

---

Centre de Recherche en Automatique de Nancy (CRAN)

Faculté des Sciences et Technologies, B.P. 70239, 54506 Vandœuvre-lès-Nancy









UNIVERSITÉ  
DE LORRAINE

FACULTE DES SCIENCES & TECHNOLOGIES

CENTRE DE RECHERCHE  
EN AUTOMATIQUE DE  
NANCY



CNRS UMR 7039

U.F.R. : Sciences et Techniques Mathématiques, Informatique et Automatique

Ecole Doctorale : IAEM Lorraine

Département de Formation Doctorale : Automatique

Thèse  
présentée pour l'obtention du titre de  
Docteur de l'Université de Lorraine  
en Sciences, spécialité Automatique,  
Traitement du Signal et Génie Informatique  
par **Makkaoui** LEILA

# Compression d'images dans les réseaux de capteurs sans fil

Soutenue publiquement le 26 Novembre 2012

Membres du jury :

<i>Rapporteurs :</i>	M. Jean-Jacques Pansiot	Professeur, Université de Strasbourg
	M. Amine Nait-Ali	Professeur, Université Paris-Est Créteil
<i>Examineurs :</i>	M. Francis Lepage	Professeur, Université de Lorraine
	M. Congduc Pham	Professeur, Université de Pau
	M. Jean-Marie Moureaux	Professeur, Université de Lorraine (Directeur de thèse)
	M. Vincent Lecuire	MdC, Université de Lorraine (Co-Directeur de thèse)

---

Centre de Recherche en Automatique de Nancy (CRAN)

Faculté des Sciences et Technologies, B.P. 70239, 54506 Vandœuvre-lès-Nancy





À mes parents,  
mes soeurs,  
mon beau frère,  
Mehdi et Ilyes

# Remerciements

Ce travail a été effectué au sein du Centre de Recherche en Automatique de Nancy (CRAN, UMR 7039 du CNRS) dirigé par le Professeur Alain RICHARD que je remercie vivement pour son accueil au sein du laboratoire. Plus précisément, ma thèse a été réalisée au sein de l'équipe Système Contrôlés en Réseau du groupe thématique SYMPA (Systèmes de Production Ambiants) animé par le Professeur Thierry DIVOUX que je remercie également pour son accueil et sa convivialité.

Je tiens dans un premier temps à remercier le Professeur Jean-Marie MOUREAUX pour m'avoir confié ce travail de recherches, ainsi que ses précieux conseils au cours de ces années. Je remercie également Monsieur Vincent LECUIRE, Maître de conférences de l'université de Lorraine, co-encadrant de ce travail de thèse, pour sa disponibilité, ses idées et conseils, ainsi que pour son aide précieuse de tous les jours.

Je tiens à remercier Monsieur Amine NAIT-ALI, Professeur à l'université Paris-Est Créteil, et Monsieur Jean-Jacques PANSIOT, Professeur à l'Université de Strasbourg, d'avoir accepté d'être rapporteurs de cette thèse. Je souhaite adresser également mes remerciements à Monsieur Francis LEPAGE, professeur à l'université de Lorraine et Monsieur Congduc PHAM d'avoir fait l'honneur d'examiner mon travail.

J'adresse mes remerciements à toute l'équipe du département Génie Electrique et Informatique Industrielle : les enseignants-chercheurs et mes collègues doctorants pour tout les échanges techniques, scientifiques et pour leur sympathie, leur accueil chaleureux pendant ces années de thèse.

J'ai pu travaillé dans un cadre particulièrement agréable, grâce à l'ensemble des membres de l'équipe SYMPA. Je pense particulièrement à Esma et sa petite famille qui m'a énormément aidé pendant ces années, à Idriss pour ses conseils, à Cheick pour nos nombreuses discussions, à Sylvain et Jérémy pour leur aide, à Saïd pour son soutien et son encouragement. Je tiens à remercier Mario, Thomas, Nicolas, Gilbert, David, Alexandre, Pierre, Pascale, Lassaad, Boutheina. Merci à tous pour votre bonne humeur, pour toutes ces séances de rires et sourires, et pour toutes ces discussions autour d'un café.

Cette thèse, aboutissement de longues années d'études, je la dois beaucoup à mes soeurs : Raja, Naouel, Raoudha et Olfa. Merci à mes parents exceptionnels qui m'ont toujours encouragé et donné de la force pour persévérer et pour prospérer dans la vie. Il m'est impossible de trouver des mots pour leur remercier pour tout ce qu'ils ont fait.

# Table des matières

<b>Introduction</b>	<b>xi</b>
0.1 Contexte et objectifs . . . . .	xi
0.2 Organisation du document . . . . .	xiii
<b>1 Le problème de l'énergie dans les réseaux de capteurs sans fil</b>	<b>1</b>
1.1 Introduction . . . . .	1
1.2 Caractéristiques des réseaux de capteurs . . . . .	4
1.2.1 Anatomie d'un nœud capteur . . . . .	5
1.2.2 Modèles de fonctionnement du réseau . . . . .	7
1.2.3 Problème de la consommation d'énergie . . . . .	10
1.2.4 Durée de vie d'un réseaux de capteurs . . . . .	13
1.3 Techniques de conservation de l'énergie . . . . .	14
1.3.1 Réveil cyclique (Duty-cycling) . . . . .	14
1.3.2 Les protocoles de routage efficaces en énergie . . . . .	18
1.3.3 Contrôle de la topologie . . . . .	21
1.3.4 Techniques de réduction des données . . . . .	22
1.4 Le cas des réseaux de capteurs d'images . . . . .	25
1.4.1 Exemples d'application . . . . .	27
1.4.2 Travaux relatifs aux réseaux de capteurs d'images . . . . .	28
1.5 Conclusion et Perspectives . . . . .	33
<b>2 Compression d'images et contributions</b>	<b>35</b>
2.1 Généralités . . . . .	36
2.2 Les techniques de base de la compression d'images . . . . .	38
2.2.1 Techniques de décorrélation intra-image . . . . .	38
2.2.2 Quantification . . . . .	44
2.2.3 Codage entropique . . . . .	47
2.2.4 Synthèse . . . . .	51

2.3	Réduction DCT . . . . .	51
2.3.1	L'approche par factorisation de la matrice (DCT rapide) . . . . .	53
2.3.2	L'approche zonale (DCT réduite) . . . . .	58
2.3.3	La DCT zonale rapide . . . . .	60
2.3.4	Evaluation des performances . . . . .	73
2.4	Conclusion . . . . .	76
<b>3</b>	<b>DCT zonale rapide : choix de la taille de la zone en fonction du débit</b>	<b>79</b>
3.1	Introduction . . . . .	79
3.2	Les distorsions . . . . .	80
3.2.1	Les distorsions liées à la compression . . . . .	80
3.2.2	Autres dégradations . . . . .	81
3.3	Mesures de la qualité des images reçues . . . . .	81
3.3.1	Les approches de type purement signal . . . . .	82
3.3.2	Les approches structurelles . . . . .	84
3.3.3	PSNR-HVS . . . . .	88
3.4	Expérimentation . . . . .	88
3.4.1	Bases d'évaluation . . . . .	88
3.5	Évaluation des performances . . . . .	92
3.6	Conclusion . . . . .	99
<b>4</b>	<b>Évaluation des performances</b>	<b>105</b>
4.1	Introduction . . . . .	105
4.2	Evaluation par analyse numérique . . . . .	106
4.3	Évaluation par expérimentation sur un nœud réel . . . . .	109
4.3.1	Présentation de la plateforme utilisée . . . . .	109
4.3.2	Banc d'essai utilisé pour les mesures . . . . .	111
4.3.3	Estimation durée de vie d'un nœud capteur . . . . .	112
4.4	Implantation de la chaîne de compression sur circuit ASIC . . . . .	115
4.5	Conclusion . . . . .	118
	<b>Conclusions et perspectives</b>	<b>119</b>
	<b>Liste des publications</b>	<b>123</b>
	<b>Notice bibliographique</b>	<b>125</b>

# Table des figures

1.1	Schéma conventionnel d'un réseau de capteurs sans fil . . . . .	2
1.2	Les composants d'un capteur sans fil . . . . .	5
1.3	Techniques de conservation d'énergie existantes (Anastasi <i>et al.</i> , 2009) . . . . .	15
2.1	Schéma général de compression avec perte . . . . .	38
2.2	Transformée en ondelettes 2D . . . . .	41
2.3	Image résultante après le premier niveau décomposition . . . . .	41
2.4	Image Lena originale $512 \times 512$ . . . . .	42
2.5	Décomposition DWT 1 niveau de l'image Lena $512 \times 512$ . . . . .	42
2.6	Décomposition pyramidale d'une image . . . . .	43
2.7	Décomposition DWT 2 niveaux de l'image Lena $512 \times 512$ . . . . .	44
2.8	Principe de quantificateur vectoriel . . . . .	46
2.9	Les méthodes de calcul de la DCT . . . . .	52
2.10	Graphe des flux de Chen . . . . .	56
2.11	Les symboles utilisés pour les graphes de flux . . . . .	57
2.12	Les formes de DCT zonale a) Forme carrée b) Forme triangulaire . . . . .	60
2.13	Les méthodes de calcul de la DCT zonale rapide . . . . .	60
2.14	Graphe de flux de Loeffler . . . . .	61
2.15	Graphe de flux de AAN (8 entrées, 8 sorties) . . . . .	64
2.16	Graphe de flux de AAN $k = 4$ (8 entrées, 4 sorties) . . . . .	64
2.17	Graphe de flux de Feig-Winograd . . . . .	66
2.18	Graphe de flux de BinDCT(8 entrées, 8 sorties) . . . . .	67
2.19	Graphe de flux de BinDCT $k = 4$ (8 entrées, 4 sorties) . . . . .	67
2.20	Graphe de flux de Cordic DCT . . . . .	70
2.21	Graphe de flux de Cordic Loeffler(8 entrées, 8 sorties) . . . . .	71
2.22	Graphe de flux de Cordic Loeffler (8 entrées, 4 sorties) . . . . .	71
2.23	PSNR en fonction de l'énergie : codage de Golomb+arithmétique . . . . .	74

2.24	PSNR en fonction de l'énergie : codage de Golomb+arithmétique . . . . .	75
3.1	Carte d'erreurs mathématiques . . . . .	83
3.2	Comparaison de la qualité visuelle de l'image Lena et l'image Mandrill . . .	84
3.3	carte d'erreur structurelle . . . . .	87
3.4	Images originales à niveau de gris de la base de test IVC . . . . .	90
3.5	Images originales transformées à niveau de gris de la base de test Tayoma .	91
3.6	PSNR, PSNR-HVS et SSIM de l'image Lena codée à 0,5 bpp . . . . .	93
3.7	Comparaison de la qualité visuelle de l'image Barbara . . . . .	94
3.8	PSNR, PSNR-HVS et SSIM de l'image Peppers codée à 0,5 <i>bpp</i> . . . . .	95
3.9	PSNR, PSNR-HVS et SSIM de l'image Barbara codée à 0,5 <i>bpp</i> . . . . .	95
3.10	Comparaison de la qualité visuelle de l'image Lena . . . . .	96
3.11	Valeur moyenne de $k$ en fonction de débit pour la métrique PSNR . . . . .	97
3.12	Valeur moyenne de $k$ en fonction de débit pour la métrique M-SSIM . . . . .	98
3.13	Valeur moyenne de $k$ en fonction de débit pour la métrique PSNR-HVS . . .	98
3.14	$k$ moyen de PSNR-HVS+M-MSSIM et PSNR+MSSIM . . . . .	99
3.15	Image "Lena" (512 × 512 pixels) et "Plane" (768 × 512 pixels) . . . . .	100
3.16	Image Lena, Peppers et Plane codées à 0,95 <i>bpp</i> . . . . .	101
3.17	Images d'évaluation . . . . .	102
4.1	Energie de compression et de transmission d'une image . . . . .	109
4.2	Un nœud capteur Mica2 . . . . .	110
4.3	Une caméra Cyclops attachée à un mote Mica2 . . . . .	110
4.4	Banc d'essai utilisé pour les mesures. . . . .	112
4.5	Trace de la puissance consommée par le capteur . . . . .	113
4.6	Architecture en bloc du circuit de compression . . . . .	116
4.7	Dessin de masque du circuit type JPEG proposé . . . . .	117

# Introduction

## 0.1 Contexte et objectifs

Les réseaux de capteurs font référence à des réseaux sans fil spontanés (structure émergeant d'interaction entre les nœuds suivant les principes d'auto-organisation et d'auto-stabilisation) dédiés à l'observation de phénomènes dynamiques complexes : Les nœuds-capteurs sont tous petits dispositifs électronique autonomes, alimentés par batterie et dotés de capacités de transmission sans fil (communément par ondes radio), qui vont mesurer des grandeurs physiques et coopérer pour remonter l'information, de proche en proche, vers un point de collecte. Les réseaux de capteurs ont des atouts manifestes : ils peuvent être déployés très rapidement, couvrir de très grands espaces géographiques et fonctionner sans intervention humaine avec une grande tolérance aux défaillances.

En raison de leur flexibilité, de leur faible coût et de la facilité de déploiement, les réseaux de capteurs sans fil promettent de révolutionner notre vie à travers plusieurs domaines d'applications tels que la détection et la surveillance des désastres, le contrôle de l'environnement et la cartographie de la biodiversité, le bâtiment intelligent, l'agriculture de précision, la surveillance et la maintenance préventive des machines, la médecine et la santé, la logistique et les transports intelligents.

Pour les surveillances de longue durée, c'est-à-dire pour les réseaux dont la durée de vie escomptée est de l'ordre de plusieurs mois, voire plusieurs années, la conservation de l'énergie est un problème fondamental sachant que le remplacement des batteries des capteurs est usuellement impossible. La contrainte de l'énergie motive une grande partie des problématiques de recherche sur les réseaux de capteurs sans fil.

La consommation d'énergie des capteurs joue donc un rôle important dans la durée de vie du réseau qui est devenue un critère de performance prédominant. Plusieurs travaux de recherche gravitent autour d'un objectif commun : l'identification et la caractérisation des activités les plus consommatrices en énergie et l'optimisation de la consommation énergétique des nœuds-capteurs. Il est couramment admis que l'émetteur radio est un

des composants les plus gourmands en énergie (Raghunathan *et al.*, 2002; Shih *et al.*, 2001). Par conséquent la plupart de l'énergie dissipée par un nœud-capteur concerne la transmission et la réception des données.

Traditionnellement, les réseaux de capteurs ont été conçus pour capturer des données scalaires simples, qui peuvent être codées sur quelques octets comme la température, la luminosité, la pression atmosphérique ou les vibrations. Récemment, les progrès de la microélectronique ont permis la naissance d'une nouvelle génération de capteurs d'images miniatures ayant une très faible consommation d'énergie. Les applications utilisant ce type de capteur sont appréciables pour tout ce qui concerne la détection, le dénombrement et la localisation d'objets par la vision (comptage de la faune dans les réserves naturelles, localisation des départs de feu en forêt, etc.). Mais elles sont particulièrement gourmandes en énergie puisque les nœuds engagent des volumes de données très largement supérieurs aux mesures scalaires classiques. En effet, alors que pour le codage d'une mesure de température, 2 ou 3 octets sont largement suffisants (ce qui peut être contenu dans un seul paquet) une image est généralement représentée sur plusieurs milliers d'octets en fonction de sa taille et sa résolution. Par conséquent, le nœud-capteur va devoir générer beaucoup de paquets pour transmettre l'image entière, et donc consommer beaucoup d'énergie. Il semble évident que le coût d'énergie de la transmission de l'image peut être réduit significativement en compressant l'image à la source. Bien que les méthodes de compression existantes fournissent d'excellentes performances en termes de rapport débit-distorsion, elles ne sont pas éligibles en l'état dans le contexte des réseaux de capteurs car elles 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 *et al.*, 2005).

Afin de prolonger la durée de vie du capteur d'image et du réseau tout entier, le développement de méthodes de compression tenant compte de l'efficacité énergétique est nécessaire pour que le déploiement de réseaux de capteurs d'images puisse être envisagé en pratique.

La présente thèse forme une contribution à l'efficacité énergétique des méthodes de compression d'images dans les réseaux de capteurs sans fil. Notre objectif est de proposer et de concevoir une chaîne de compression d'image qui fournisse un compromis entre l'énergie consommée par les capteurs et la qualité visuelle des images. Nous nous intéressons spécialement à la dépense d'énergie sur les nœuds-caméras même si la durée de vie du réseaux n'en dépend qu'en partie.



## 0.2 Organisation du document

Ce manuscrit est organisé en quatre chapitres suivis d'une section donnant les conclusions et les perspectives du travail. Dans le chapitre 1, nous présentons un état de l'art sur la conservation de l'énergie dans les réseaux de capteurs sans fil. Nous commençons d'abord par présenter les caractéristiques de fonctionnement générales des réseaux de capteurs, indépendamment du type des informations collectées. Nous décrivons aussi les principales techniques permettant d'améliorer leur efficacité énergétique. Ensuite, nous abordons le cas spécifique des réseaux de capteurs d'images et nous faisons un état de l'art du domaine. Cette section permet de positionner notre contribution au problème de la conservation de l'énergie par rapport aux travaux faisant référence dans les domaines de la compression et de la transmission d'images.

Le chapitre 2 constitue le coeur de notre travail de thèse : Nous présentons une méthode de compression d'images pour les réseaux de capteurs sans fil qui, d'une part, est très peu calculatoire donc peu gourmande en énergie, et qui, d'autre part, assure une bonne qualité visuelle de l'image restituée après décompression. L'algorithme proposé est basé sur la transformée en cosinus discrète (DCT) et consiste à opérer sur un sous-ensemble limité de coefficients situés dans une zone du bloc  $8 \times 8$ . Elle permet de sélectionner les coefficients les plus importants pour coder les informations significatives localisées dans ce bloc de l'image. La qualité de l'image reçue est contrôlée par la dimension de la zone de sélection des coefficients (paramètre  $k$ ).

Dans le chapitre 3, nous évaluons l'impact de la valeur du paramètre  $k$  sur la qualité visuelle des images, en considérant différentes métriques de mesure de qualité. Nous déterminons également la valeur optimale du paramètre  $k$  pour des plages de débit donnés. L'utilisateur final pourra alors choisir la valeur de  $k$  selon les exigences de l'application.

Dans le chapitre 4, nous évaluons analytiquement puis expérimentalement la consommation d'énergie de notre chaîne de compression lorsqu'elle est implantée logiquement sur un nœud-capteur réel. Notre plateforme était composée d'un nœud Mica2 et d'une caméra Cyclops. Nous analysons son impact sur la durée de vie du nœud-caméra. Nous présentons ensuite les performances de la chaîne de compression lorsqu'elle est implantée sur un circuit intégré suivant la technologie ASIC.



# Chapitre 1

## Le problème de l'énergie dans les réseaux de capteurs sans fil

### Sommaire

---

<b>1.1</b>	<b>Introduction</b>	<b>1</b>
<b>1.2</b>	<b>Caractéristiques des réseaux de capteurs</b>	<b>4</b>
1.2.1	Anatomie d'un nœud capteur	5
1.2.2	Modèles de fonctionnement du réseau	7
1.2.3	Problème de la consommation d'énergie	10
1.2.4	Durée de vie d'un réseaux de capteurs	13
<b>1.3</b>	<b>Techniques de conservation de l'énergie</b>	<b>14</b>
1.3.1	Réveil cyclique (Duty-cycling)	14
1.3.2	Les protocoles de routage efficaces en énergie	18
1.3.3	Contrôle de la topologie	21
1.3.4	Techniques de réduction des données	22
<b>1.4</b>	<b>Le cas des réseaux de capteurs d'images</b>	<b>25</b>
1.4.1	Exemples d'application	27
1.4.2	Travaux relatifs aux réseaux de capteurs d'images	28
<b>1.5</b>	<b>Conclusion et Perspectives</b>	<b>33</b>

---

### 1.1 Introduction

Les progrès récents dans les domaines de la microélectronique, des communications sans fil et de l'informatique embarquée ont permis de fabriquer des systèmes de mesure autonomes de plus en plus petits, appelés nœuds-capteurs ou tout simplement capteurs, capables de mesurer une ou plusieurs grandeurs physiques, et de stocker, traiter et communiquer les données. Suivant des principes similaires aux réseaux mobiles ad hoc, les nœuds-capteurs vont découvrir leurs voisins et s'auto-organiser pour former un réseau de

capteurs sans fil. Un tel réseau spontané, sans infrastructure préétablie, offre un moyen élégant pour remonter les données vers l'utilisateur final sur la base d'un modèle de communication multi-sauts, comme schématisé FIGURE 1.1. Par principe, les nœuds-capteurs ont donc (au moins) deux rôles : ce sont des sources d'information (acquisition des mesures) et des relais intermédiaires de son transport de bout en bout (transmission de proche en proche pour l'acheminement des paquets jusqu'à destination). Le point de collecte, appelé puits ou station de base, fait l'interface entre le réseau de capteurs sans fil et l'utilisateur. L'interconnexion du puits à un réseau à infrastructure (e.g., Internet par liaison satellite) permet les accès à distance pour les utilisateurs éloignés.

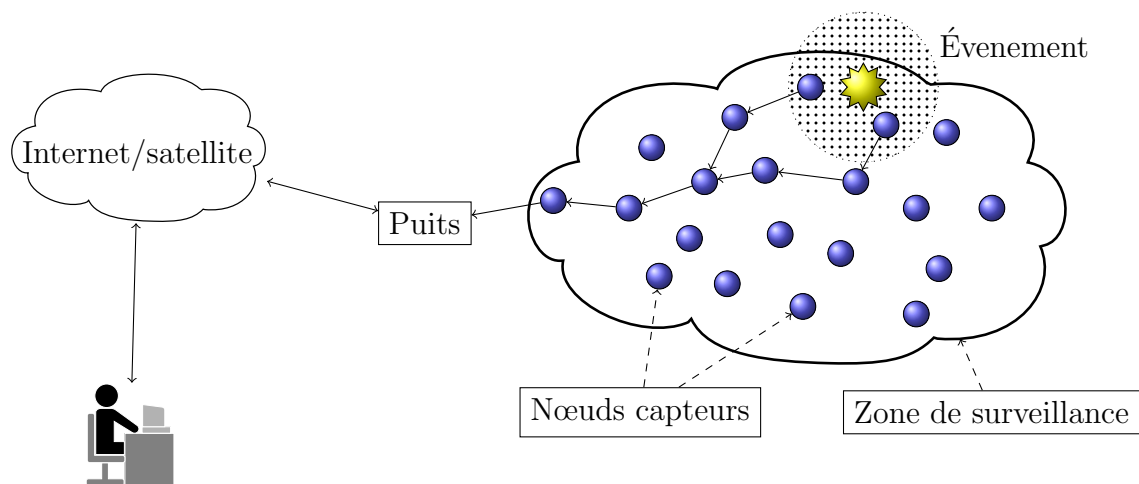


FIGURE 1.1: Schéma conventionnel d'un réseau de capteurs sans fil

Le magazine Technology Review publié par le MIT présentait dans son numéro de février 2003 (MIT, 2003) les réseaux de capteurs sans fil comme l'une des dix technologies qui vont changer le monde. Il devient en effet possible d'observer des phénomènes dynamiques complexes (physiques, chimiques ou biologiques) sur de très grands espaces géographiques, juste en déployant en nombre ces capteurs minuscules. Le réseau de capteurs se forme automatiquement et s'adapte aux changements d'état des nœuds. Il est donc capable de fonctionner sans aucune intervention humaine avec une grande tolérance aux défaillances, être déployé de manière aléatoire et opérer en milieu hostile ou difficile d'accès. Les applications potentielles des réseaux de capteurs sont extrêmement variées (Estrin *et al.*, 2001; Akyildiz *et al.*, 2002a). Citons à titre indicatif des applications du domaine militaire (surveillance tactique terrestre et maritime, cartographie et pistage de cibles), de la surveillance environnementale (études climatiques, suivi de la pollution des sols, de l'eau, de l'air, agriculture de précision) et de la prévention des risques (feux de forêts, inondations, activité sismique ou volcanique, structure des grands ouvrages). Dans

la majorité des applications, les ondes radio sont évidemment considérées pour la communication entre capteurs. D'autres techniques de transmission sont toutefois possibles, notamment les ondes acoustiques qui sont bien appropriées dans les milieux sous-marins (Dong, 2009).

Les réseaux de capteurs sans fil amènent des défis scientifiques et technologiques couvrant des disciplines comme la microélectronique (électronique basse consommation, émetteurs radio nanowatt, systèmes sur puce, mémoires magnéto-résistives), l'automatique (diagnostic des systèmes dormants, observateurs distribués, commande par consensus), le traitement du signal (compression de source distribué, échantillonnage compressé, géolocalisation), l'informatique et les réseaux (qualité de service, protocoles robustes au facteur d'échelle, masse de données, composants logiciels reconfigurables), la physique-chimie (transducteurs, mécanismes de récupération d'énergie à haut rendement) et la biologie (biopiles, systèmes bio-inspirés). Un problème fondamental, et transverse à toutes ces disciplines, concerne la conservation de l'énergie, étant donné que les nœuds capteurs sont alimentés par batterie et que leur remplacement est bien souvent impossible. Cela devient un véritable challenge pour les surveillances sur de très longues durées, de l'ordre de plusieurs mois, voire de plusieurs années. La miniaturisation continue des composants électroniques couplée à la diminution de leur consommation électrique va bien sûr contribuer à augmenter l'autonomie énergétique des nœuds-capteurs. Toutefois, le problème de la conservation de l'énergie n'est pas seulement un problème d'électronique, c'est aussi un problème d'ordonnancement du temps d'activité des nœuds-capteurs pour conserver la connexité du réseau et la couverture de la zone d'intérêt sur la durée escomptée. Autrement dit, c'est la durée de vie du réseau tout entier qui compte, pas celle des nœuds-capteurs pris individuellement. Cette assertion est facile à démontrer par l'exemple, le plus symbolique étant le phénomène du trou d'énergie (Li et Mohapatra, 2005; Lian *et al.*, 2006) : il prend pour hypothèse (vérifiée en pratique) que l'émetteur radio est un des composants les plus gourmands en énergie. Avec un modèle de communication multi-sauts convergeant vers le puits, plus un nœud est proche du puits, plus il agrège de trafic, et donc plus il s'épuise rapidement. Dans le cas d'un réseau de capteurs de densité uniforme, des simulations (Lian *et al.*, 2006) ont montré que seulement 10% de l'énergie initialement disponible dans le réseau avait été consommée quand le puits devient inatteignable.

Parmi toutes les applications potentielles des réseaux de capteurs sans fil, celles utilisant des capteurs d'image sont appréciables pour tout ce qui concerne la reconnaissance, la localisation et le dénombrement d'objets par la vision (Soro et Heinzelman, 2009). Dans ce cas, les nœuds ou une partie des nœuds du réseau sont équipés de mini-caméras à tech-

nologie CMOS. Les réseaux de capteurs d'image amplifient le problème de la conservation de l'énergie puisque le volume de données engagé par échantillon est de plusieurs ordres de grandeur supérieur à une valeur scalaire classique (e.g., une mesure de température). Ce chapitre vise à présenter un état de l'art des travaux traitant de la conservation de l'énergie dans ce type de réseau particulier. Il est structuré en trois parties. La section 1.2 présente rapidement les caractéristiques de fonctionnement générales aux réseaux de capteurs, indépendamment du type des informations collectées. La section 1.3 aborde les principales techniques permettant d'améliorer leur efficacité énergétique, quelque soit le type de la capture, et qui peuvent donc être utilisés dans le cas des capteurs d'images. La section 1.4 traite le cas particulier des réseaux de capteurs d'image et fait un état de l'art détaillé des contributions sur le problème de l'énergie propre à ce type de réseau. Cette section permet de positionner notre contribution au problème de la conservation de l'énergie par rapport aux travaux faisant référence en compression et en transmission d'images.

## 1.2 Caractéristiques des réseaux de capteurs

Les réseaux de capteurs sont des réseaux dédiés à la surveillance de grands espaces, généralement sur de longues durées, pour observer et analyser finement des phénomènes dynamiques complexes. Leurs caractéristiques sont donc très différentes des réseaux sans fil traditionnels :

Tout d'abord, les nœuds capteurs sont par nature des systèmes peu sophistiqués (ils sont conçus pour accomplir une tâche) et très limitée en capacité de calcul, de mémoire et de communication. Cette limitation des ressources a plusieurs explications : la première est liée à la contrainte de coût de fabrication des nœuds capteurs. Ce coût doit rester réduit pour que les applications engageant des réseaux de plusieurs milliers, voire millions de nœuds, deviennent une réalité. Un coût à l'unité inférieur à 1 *US\$* fait référence dans la littérature. La deuxième est liée à la contrainte de l'énergie. Privilégier les circuits électroniques de très faible consommation électrique se fait au détriment des "performances". La dernière est liée à la contrainte de poids et de taille, en particulier dans le challenge de fabriquer des microcapteurs ( $< 1mm^3$ ) et, à termes, de la poussière de capteurs (échelle nanoscopique).

Ensuite, les nœuds-capteurs sont typiquement conçus pour un usage unique, c'est à dire que leurs fonctionnement va dépendre de l'application pour laquelle ils sont destinés. Pour ce qui traite de la communication il n'y a pas une pile de protocoles qui puisse faire

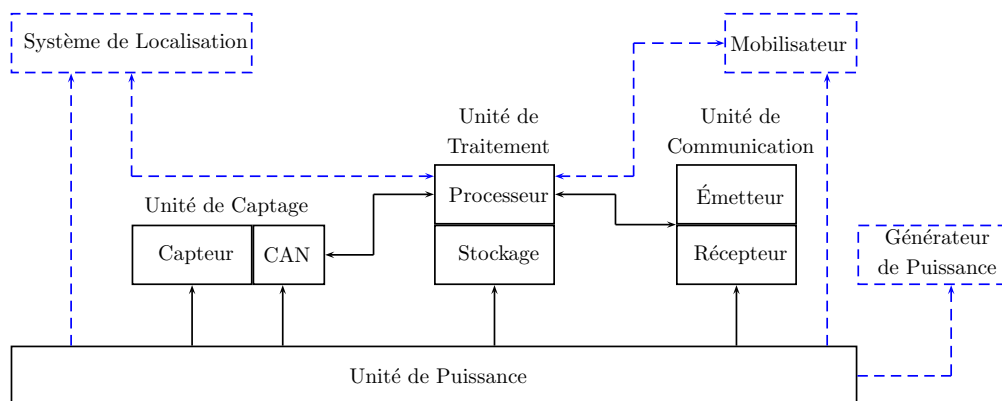


FIGURE 1.2: Les composants d'un capteur sans fil

référence pour l'ensemble des applications potentielles.

Nous nous bornerons donc, dans cette section, à présenter les caractéristiques les plus classiques des réseaux de capteurs, des nœuds capteurs, des modèles de déploiement du réseau, des modèles de collecte des données sachant qu'il n'est pas possible d'être exhaustif.

### 1.2.1 Anatomie d'un nœud capteur

Un capteur sans fil a pour fonction, à la base, de mesurer des grandeurs relatives à un phénomène physique, de stocker si besoin les données de mesure ainsi que de transmettre les données brutes ou pré-traitées à l'utilisateur final.

Il est donc composé de quatre module élémentaires : une unité d'acquisition, une unité de traitement et une unité de communication ainsi que d'une unité d'alimentation par batterie. Il peut aussi intégrer, c'est optionnel, une unité de géolocalisation et une unité de récupération d'énergie. L'anatomie d'un capteur est schématisée FIGURE 1.2 (Akyildiz *et al.*, 2002b).

**L'unité d'acquisition :** Elle prend en charge la perception des phénomènes à observer et de la conversion des signaux analogiques en valeurs numériques. Cette unité comprend donc le ou les capteurs proprement dit (e.g., température, lumière ...) et des convertisseurs analogiques numériques.

**L'unité de traitement :** C'est le cœur de système. Elle commande les autres composants du nœud capteur selon un scénario programmé, notamment l'unité d'acquisition (relevé de mesure) et l'unité de communication (envoi et réception des messages). Elle prend en charge le traitement, l'exploitation et le stockage des données,

y compris celles relatives aux protocoles de communication des couches hautes (gestion des tables de voisinage, de routage, encodage et décodage des paquets). Dans sa forme courante, elle est constituée d'un microcontrôleur et de mémoire.

Le microcontrôleur est caractérisée typiquement par une basse consommation d'énergie (de l'ordre de 1  $mW$ ) et une faible cadence de fonctionnement (moins de 10  $MHz$ ). La quantité de mémoire est de l'ordre de 10  $Ko$  de RAM pour les données et de 10  $Ko$  de ROM pour les programmes (Karl et Willig, 2005).

**L'unité de communication :** Elle prend en charge le protocole de communication dépendant de la technologie sans fil utilisée, c'est à dire tout ce qui traite de la transmission de messages entre nœuds-capteurs directement à portée : le contrôle d'accès au médium, la mise en forme des trames, l'encodage et le décodage des signaux, etc. D'autres types de technologies sont toutefois possible. Citons celles par ondes acoustiques, principalement pour les surveillances sous marines (Park et Rodoplu, 2007), et plus rarement du surface (SAW) (Jatlaoui, 2009). Les technologies de communication optiques sont plus rares en raison des contraintes liées au déploiement (comme dans le cas des nœuds *smart dust* (Warneke *et al.*, 2001; Barakat, 2005)).

**L'unité d'alimentation :** Elle est responsable de la fourniture d'énergie pour tous les composants du nœud capteur. Il s'agit d'une batterie de capacité finie. Quand la batterie est épuisée, il faut la remplacer sinon le nœud cesse de fonctionner. Pour les applications (et elles sont nombreuses) où le remplacement de la batterie n'est pas souhaitable ou tout simplement impossible, la durée de vie d'un nœud est limitée à la durée de vie de sa batterie.

Remarquons que l'évolution des performances des batteries ne suit pas celles des autres composants électroniques. À titre de comparaison, la densité de la mémoire a augmenté plus de 1200 fois depuis 1990, alors que celle d'une batterie n'a augmenté que de 3 fois (Starner, 2003). Autrement dit, on ne sait pas encore fabriquer des batteries à la fois de très petite taille et de très grande capacité. C'est un verrou scientifique important.

**L'unité de géolocalisation :** Pour pouvoir exploiter convenablement les données fournies par le réseau de capteur, l'utilisateur a bien évidemment besoin de connaître la position de chaque nœud capteur. Quand le déploiement des capteurs se fait manuellement, la carte du réseau peut être préétablie et la position des nœuds préenregistrée. L'unité de géolocalisation consiste à doter le nœud capteur d'une instrumentation lui permettant d'identifier automatiquement sa position absolue



dans l'espace (comme la technique de GPS (Caruso *et al.*, 2005)) ou relative par rapport aux nœuds voisins (technique de triangulation basée sur l'estimation de distance à partir de l'atténuation des ondes radio, des délais de propagation des ondes sources, etc).

**L'unité de récupération d'énergie** : son rôle est d'augmenter la durée de vie du nœud-capteur en le dotant d'un dispositif pour recharger sa batterie à partir de sources d'énergie ambiantes, par exemple l'énergie provenant des vibrations, de la chaleur, de la lumière, et même des ondes électromagnétiques, et la convertir en courant électrique. Les auteurs de (Fuller et Millett, 2011) ont fait la comparaison de la durée de vie de la densité de puissance volumique entre des batteries et des sources d'énergies ambiantes et nous pouvons remarquer que la densité de puissance des piles diminue avec le temps, alors que la puissance des sources ambiantes présentes dans l'environnement est constante en fonction du temps.

## 1.2.2 Modèles de fonctionnement du réseau

On trouve plusieurs méthodes pour classer les réseaux de capteurs (Tilak *et al.*, 2002; Karl et Willig, 2007). Pour chaque type d'application, les réseaux de capteurs ont des caractéristiques différentes. Ils se distinguent, principalement, par le modèle de collecte des données, le modèle de déploiement des capteurs et le modèle de mobilité des nœuds.

### Modèles de collecte des données

Selon les interactions entre le réseau de capteurs et la station de base, nous citons trois modèles principaux :

- Surveillance périodique : Tous les capteurs envoient périodiquement leurs mesures à la station de base. Le type d'application visé concerne les applications de surveillance où le but principal est d'avoir une information régulière de la zone surveillée comme par exemple l'humidité du sol (soil moisture sensors project)(Quails *et al.*, 2001).
- Surveillance à la demande : Les capteurs ne transmettent pas spontanément les données de mesure, ils les envoient seulement lorsqu'ils reçoivent des requêtes explicites de la station de base. Ces requêtes peuvent être dirigées (solicitation d'un capteur ou un groupe de capteurs) ou diffusées (requête de type conditionnel, par exemple tous les capteurs dont la mesure dépasse un seuil).
- Surveillance sur déclenchement d'événements : Dans ce modèle, les capteurs envoient les mesures seulement lorsqu'il y a un événement qui se produit, généralement le

dépassement d'un seuil. Ce modèle est bien adapté par exemple à la détection d'intrusion.

- Applications hybrides : toute alliance des modèles de fonctionnements précédents.

## Modèles de déploiement des capteurs

Le déploiement des capteurs est la première opération dans le cycle de vie d'un réseau de capteurs. Le modèle de déploiement se définit par plusieurs paramètres, notamment le nombre de nœuds, la densité de déploiement et la méthode de déploiement : déterministe ou aléatoire.

- *Le déploiement déterministe* : lorsque la zone de surveillance est accessible et peu étendue, il est possible de positionner les nœuds à des emplacements préétablis. La position de chaque nœud est donc connue d'avance. Le rôle de chacun d'eux, et leurs interactions peuvent être programmés en fonction de leur position. C'est le cas par exemple pour les capteurs chargés de réguler la climatisation d'un immeuble ou de surveiller les constantes médicales de personnes malades. Ce type de déploiement peut devenir très contraignant, voire impossible pour les réseaux de capteurs à grande dimension.
- *Le déploiement aléatoire* : l'utilisation des capteurs dans des zones inaccessibles ou sensibles rend impossible un déploiement déterministe, au cours duquel chaque objet serait placé à une position prédéterminée. Les nœuds peuvent alors être déployés à l'aide de moyens divers. Dans les surveillances militaires par exemple, les capteurs pourraient être largués sur le terrain depuis un avion survolant la zone hostile. Il a souvent été question d'un déploiement aléatoire des capteurs, effectué comme un jeté de graines. en raison soit de l'hostilité de la zone à surveiller, soit de son immensité.

Le déploiement peut être fait d'un seul coup ou bien se faire de manière continue en rajoutant de nouveaux capteurs dans une même zone après des défaillances dues à l'épuisement des batteries pour prolonger la durée de vie du réseau.

Ce type de déploiement suppose que ces capteurs soient dotés d'une unité de géolocalisation.

Le déploiement aléatoire ne peut pas fournir une distribution uniforme sur la région d'intérêt, ce qui amène de nouveaux problèmes dans les réseaux de capteurs pour garantir la couverture de la zone et la connexité du réseau.

On peut distinguer trois sortes de couverture. La couverture "clairsemée", où les nœuds déployés assurent la couverture d'une partie de la zone d'intérêt seulement. La couverture "dense", dans ce cas la zone est presque complètement couverte. Enfin, La couverture "redondante", où on peut trouver des zones couvertes plusieurs fois par plusieurs nœuds.

Les couvertures redondantes servent à prolonger la durée de vie du réseau en mettant en sommeil les capteurs redondant tout en assurant la couverture de la zone surveillée et la connectivité du réseau. Ces capteurs redondants prendront le relais des capteurs arrivant au terme de leurs provision énergétique (Gallais *et al.*, 2008).

La couverture peut être considérée comme une des métriques de la qualité de service du réseau. Une mauvaise répartition des capteurs peut rendre certains nœuds ou groupes de nœuds inutilisables s'ils ne sont pas capables de communiquer jusqu'au puits.

### Modèles de mobilité

On peut distinguer trois formes de mobilité dans les réseaux de capteurs (Karl et Willig, 2007) :

– *Mobilité des nœuds* :

La mobilité des capteurs ne concerne qu'un petit nombre d'applications, par exemple des applications militaires (les capteurs sont attachés à des soldats), des applications de surveillance des animaux d'élevage, ou des oiseaux migrateurs (les capteurs sont portés par les animaux) ou encore des surveillances des milieux marins (les capteurs suivent le mouvement du courant) (Cui *et al.*, 2006).

Face à la mobilité des nœuds, le réseau doit se réorganiser assez fréquemment pour pouvoir fonctionner correctement. Le problème de la conservation de l'énergie des nœuds devient aussi plus ardu car les techniques de géolocalisation coûtent de l'énergie et elles vont devoir être sollicitées plus fréquemment (Anastasi *et al.*, 2008).

- *Mobilité des puits* : C'est un cas spécial de mobilité où seul le point de collecte est mobile, par exemple un agriculteur travaillant dans son champs. Cette forme de mobilité peut aussi être utilisée comme stratégie pour réduire la consommation d'énergie des réseaux de capteurs. Dans le principe où le trafic converge vers le puits, les nœuds proches du puits transmettent plus de paquets que les autres nœuds (Becker *et al.*, 2009) et donc épuisent plus vite leurs batteries (*funneling effect*) (Li et Mohapatra, 2007). Si ce puits est mobile, les nœuds-capteurs peuvent attendre son passage pour lui envoyer leurs données directement ou sur un petit nombre de sauts. Cela va réduire considérablement le trafic dans le réseau. Pour les réseaux de grande échelle, l'utilisation de plusieurs puits mobiles devient intéressante. (Ben Saad *et al.*,

2010), ont proposé un algorithme qui optimise le positionnement des puits mobiles pour limiter la distance moyenne entre les capteurs et le puits le plus proche.

- *Mobilité de l'événement* : Ce type de mobilité existe essentiellement dans les applications de suivi de cibles. Dans ce type d'application, il est habituellement important que l'événement observé soit couvert par un nombre suffisant de nœuds. Par conséquent, les nœuds vont se réveiller autour de l'objet, pour le surveiller avec un taux d'activité élevée, et puis entrer en sommeil lorsque la cible s'éloigne. Pendant que la source d'événement se déplace à travers le réseau, elle est accompagnée d'un secteur d'activité dans le réseau qui la suit.

### 1.2.3 Problème de la consommation d'énergie

La consommation d'énergie est un problème fondamental lorsque les capteurs sont déployés dans des zones inaccessibles ou encore déployés sur de grands espaces, c'est-à-dire lorsqu'il est difficile voire impossible de remplacer les batteries des nœuds quand elles arrivent à épuisement. De ce fait, la durée de vie limitée des nœuds va avoir un impact sur la durée de vie du réseau tout entier. Augmenter la durée de vie du réseau sans augmenter la capacité des batteries signifie réduire la dépense d'énergie des nœuds. Pour les applications de surveillance exigeant que le réseau de capteurs soit opérationnel pendant une très longue durée, économiser l'énergie est un véritable challenge. La littérature scientifique traitant de l'efficacité énergétique dans les réseaux de capteurs est abondante. Avant de présenter les techniques permettant de rendre les réseaux de capteurs sans fil moins gourmands en énergie, commençons par une analyse des principales sources de dissipation d'énergie.

Rappelons quelques notions élémentaires en électricité :

La puissance électrique est le produit de la tension électrique aux bornes du système électronique (ou dipôle) et de l'intensité électrique qui le traverse. Si la tension et le courant varient, la puissance est donnée par le formule 1.1.

$$P(t) = u(t) \times i(t) \quad (1.1)$$

avec  $P(t)$  : la puissance instantanée, exprimée en watts,  $u(t)$  la tension instantanée aux bornes du dipôle, exprimée en volts, et  $i(t)$  le courant instantané qui le traverse, exprimé en ampères.

La puissance électrique représente la quantité d'énergie par unité de temps fournie par le générateur (ici la batterie) et consommée par le système (ici l'ensemble des composants

électroniques du nœud-capteur). L'énergie électrique consommée, exprimée en Joules, est donc égale à l'intégrale de la puissance par rapport au temps :

$$E(t) = \int_0^t P(t) = \int_0^t u(t) \times i(t) \quad (1.2)$$

Tous les composants électroniques d'un capteur sans fil (aussi bien les composants passifs comme les résistances, les condensateurs, les diodes, que les composants actifs comme les transistors, les circuits intégrés) sont des sources potentielles de consommation d'énergie. La puissance consommée est variable dans le temps, en fonction des sollicitations de chacun des composants. Par exemple, un circuit mémoire ne consomme pas la même puissance électrique en lecture et en écriture, et le temps d'accès en lecture et écriture peuvent être différents. De même, les circuits intégrés (microcontrôleur, mémoire, émetteur radio fréquence) comportent généralement des modes de veille plus au moins profondes pour consommer extrêmement peu de puissance électrique quand ils n'ont pas d'activité à exécuter.

Evidemment, pour augmenter de manière significative la durée de vie d'un capteur sans fil, il faut s'intéresser en priorité aux composants les plus gourmands en énergie.

Nous allons passer en revue les sources de consommation d'énergie en considérant les éléments anatomiques d'un capteur sans fil.

### **L'unité d'acquisition :**

L'énergie consommée au moment de la mesure varie suivant la nature du capteur. En effet, alors que pour la capture des données scalaires simples, comme la température, vibrations, humidité ou la pression, quelques octets est largement suffisante pour leur codage (**2** ou **3 octets**), une image est généralement représenté sur plusieurs milliers d'octets (**16384 octets** pour une image de taille  $128 \times 128$  codée sur 8 *bits* par pixel).

### **L'unité de communication :**

L'unité de communication est le plus souvent constituée d'un émetteur/récepteur radio qui fournit au capteur la capacité de communiquer avec les autres au sein d'un réseau. Elle est généralement la partie la plus gourmande en énergie.

Le coût en réception est fixe mais le coût en émission dépend de la puissance d'émission du signal radio, de débit binaire, de la taille des données à transmettre, etc. . .

L'unité de communication met en oeuvre des protocoles de communication dépendant de la technologie utilisée (par exemple Wi-Fi (802.11), Bluetooth (802.15.1), ZigBee

(802.15.4), UWB (802.15.3) pour les technologies sans fil) et ils n'ont pas le même coût d'énergie.

L'utilisation inutile de l'émetteur radio provient principalement des phénomènes de surécoute (*Overhearing*), de *collisions*, d'écoute passive (*Idle listening*), les envois infructueux et les messages de contrôle.

- Le surécoute est la réception par un noeud d'une trame qui ne lui est pas destinée. L'énergie consommée pour la réception et le traitement des données de cette trame est perdue et sans aucun intérêt.
- Les collisions sont à la fois une source de dégradation des performances du réseau et de perte d'énergie. Les pertes de trames à cause des collisions forcent les noeuds à retransmettre le même paquet plusieurs fois et donc à rester actif pour le répéter et vérifier qu'il est bien reçu par la destination.
- L'écoute passive est l'attente d'une trame par le module radio. Cela arrive quand il a été demandé à un noeud d'être éveillé mais qu'il ne reçoit aucune trame et n'en transmet aucune non plus. Même si le noeud ne transmet pas et ne reçoit pas, le fait que son module radio soit activé et prêt pour recevoir consomme autant d'énergie que pour la réception.
- Les envois infructueux arrivent quand un noeud essaie de communiquer avec un autre noeud qui n'est plus accessible parce qu'il est en mode sommeil par exemple (ou hors de portée). Le noeud émetteur est en attente d'un acquittement, et il retransmet donc la même trame plusieurs fois. Il consomme de l'énergie en le faisant du fait qu'il soit resté en mode transmission et en mode réception pour l'éventuel acquittement.

### **L'unité de traitement :**

L'unité de traitement des données consomme généralement moins d'énergie que l'unité de communication. L'exemple présenté dans (Pottie et Kaiser, 2000) illustre cette disparité, en montrant que le coût énergétique nécessaire pour transmettre 1 *KB* sur une portée de 100 *m* est approximativement égal à celui nécessaire pour exécuter 3 *millions* d'instructions à une vitesse de 100 *millions* instructions par seconde (MIPS). Ce niveau peut être dépassé en fonction des circuits installés dans le noeud-capteur et des fonctionnalités requises.

Il est donc recommandé de privilégier le traitement local des données pour réduire l'activité de communication. Les noeuds capteurs doivent donc posséder des moyens de traitement local de données, tout en restant capable d'interagir avec les noeuds avoisinants.

Enfin, il est à noter que l'unité de traitement prend en charge le codage/décodage des

paquets de données et le maintien de tables diverses (pour le voisinage, le routage, etc) pour les protocoles de communication des couches hautes. La conception des algorithmes et protocoles de communication va donc influencé largement l'énergie consommée par l'unité de traitement.

#### 1.2.4 Durée de vie d'un réseaux de capteurs

La définition de la durée de vie d'un capteur sans fil est simple et sans ambiguïté : un nœud est vivant tant qu'il peut fonctionner, donc sa durée de vie est égale à celle de sa batterie. Donner une définition de la durée de vie du réseau de capteurs tout entier est moins évident. C'est pourtant un critère de performance dominant pour beaucoup d'applications.

La durée de vie d'un réseau de capteurs a fait l'objet de plusieurs définitions ciblées en fonction des besoins spécifiques des applications. Les principales définitions sont données dans (Mahfoudh et Minet, 2008) et rappelées ci-dessous.

- Durée de vie basée sur le nombre des capteurs vivants :

Cette métrique se base sur le fait que la durée de vie d'un réseaux de capteurs sans fil est égale à la durée jusqu'à ce que  $n$  nœud parmi  $m$  meurent par épuisement de batterie avec  $m$  le nombre total de nœuds du réseau. Cette durée de vie est facile à calculer car elle ne prend pas en compte le changement de la topologie réseau (Dietrich et Dressler, 2009; Tian et Georganas, 2002). Toutefois, dans les réseaux de capteurs sans fil dense, où les redondances sont présentes, cette métrique ne présente pas bien l'évaluation de la durée de vie du réseaux. Le seul cas où cette métrique pourrait raisonnablement être utilisée est dans le cas où tout les nœuds sont d'une même importance dans le réseau.

Dans (Soro et Heinzelman, 2005; Blough et Santi, 2002), les auteurs définissent la durée de vie du réseau de capteurs par la durée durant laquelle un cluster-head<sup>1</sup> épuise totalement son énergie. Cependant, dans la plupart des travaux, le cluster-head change dynamiquement quand il atteint un certain seuil d'énergie.

- Durée de vie basée sur la connectivité : (Carbunar *et al.*, 2006) définissent la durée de vie de réseau comme la durée pendant laquelle un pourcentage donné de nœuds possèdent une route vers le puits. Dans (Chiasserini *et al.*, 2006), les auteurs définissent la durée de vie de réseau comme la durée jusqu'à ce que le nombre

---

1. Le rôle du cluster-head est de faire le relais entre les nœuds du cluster et la station de base directement ou via d'autres cluster-head. Il possède généralement des ressources énergétiques supérieures aux autres nœuds du réseau.

des nœuds capteurs vivants ou connectés au réseau descend au dessous d'un seuil prédéfini.

- Durée de vie basée sur la couverture :

La zone de couverture du réseau montre si le réseau est capable de détecter un événement n'importe où dans la zone à surveiller. Avec cette métrique, la durée de vie du réseau est définie par la durée durant laquelle le réseau arrive à couvrir la zone à surveiller.

En conclusion, la durée de vie de réseau de capteurs sans fil doit tenir compte de la connectivité et de la couverture si elles sont nécessaires aux besoins des applications. La connaissance des exigences de l'application aide les concepteurs à affiner la définition de la durée de vie du réseau afin d'aboutir à une évaluation beaucoup plus réaliste et plus précise pour les utilisateurs de l'applications.

## 1.3 Techniques de conservation de l'énergie

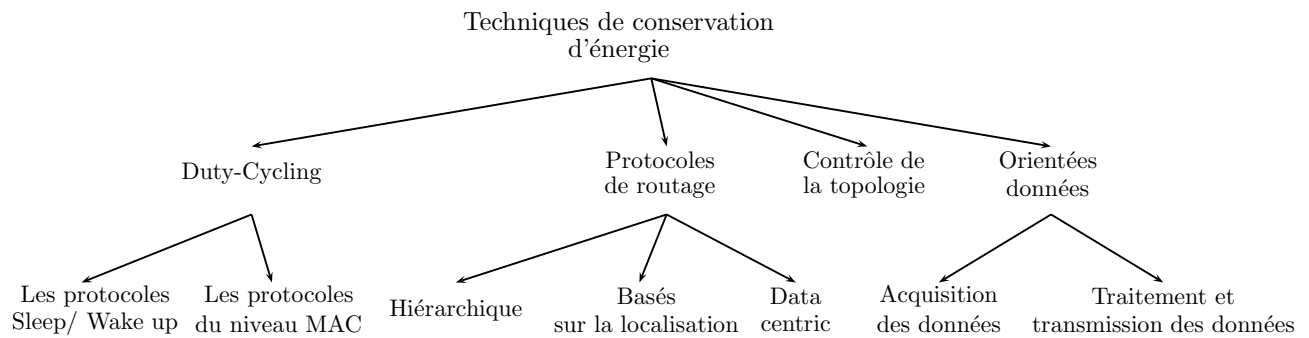
De nombreuses solutions de conservation d'énergie pour les réseaux de capteurs sans fil ont été proposées, allant de la couche physique et des techniques de modulation, jusqu'à la couche application et le développement de logiciels spécialisés (Akyildiz *et al.*, 2002b; Akyildiz *et al.*, 2007). Une classification des différentes approches de conservation d'énergie est présentée par (Anastasi *et al.*, 2009). Comme schématisé FIGURE 1.3, il y a quatre catégories : les techniques d'alternance entre les périodes d'activités et de sommeils ou "Duty cycling", les techniques basées sur les protocoles de routage, le contrôle de la topologie et les techniques orientées données basées sur l'agrégation de données « Data driven ».

### 1.3.1 Réveil cyclique (Duty-cycling)

Le moyen le plus efficace pour conserver l'énergie, est d'éteindre la radio des nœuds quand la communication n'est pas nécessaire. L'idée est que les capteurs doivent éteindre leurs radios lorsqu'il n'y a pas d'activités sur le réseau et être prêt dès qu'ils ont un paquet à recevoir ou à transmettre. Ainsi, les nœuds alternent entre périodes actives et sommeil en fonction de l'activité.

Ce comportement est dénommé "*duty-cycling*" qui est donc le passage du nœud-capteur de l'état actif à l'état sommeil et l'état sommeil à l'état actif selon l'activité du réseau. Cette technique peut être appliquée dans les réseaux à faible densité ou à forte densité. Chacun d'eux utilise des protocoles différents qui seront décrits dans les paragraphes



FIGURE 1.3: Techniques de conservation d'énergie existantes (Anastasi *et al.*, 2009)

suivantes.

### Les protocoles *Sleep/Wake up*

Dans différents types d'applications de WSN, les nœuds perdent une partie de leurs énergies durant l'état "en écoute" en attendant la détection d'un événement. Cette énergie est assez importante puisqu'elle est presque du même ordre que celle nécessaire à la réception.

Les protocoles *Sleep/Wake up* servent à limiter au maximum l'état "en écoute" en entraînant les nœuds dans un état endormi périodique. On peut relever les principaux plans sleep/wakeup implantés sous forme de protocoles indépendants au-dessus du protocole MAC (au niveau de la couche réseau ou de la couche application). Dans (Anastasi *et al.*, 2009), les auteurs présentent trois types de réveil : les réveils périodiques, les réveils actifs (dus aux déclenchements/changements des paramètres de captage par l'atteinte d'un certain seuil) et les réveils organisés (des algorithmes qui synchronisent et organisent les capteurs à réveiller pour garantir une couverture minimale du réseau).

### Les protocoles du niveau MAC

Ce type de protocole concerne les nœuds sélectionnés pour rester actif afin d'assurer la connectivité du réseau. Cependant, ces nœuds peuvent passer en mode sommeil quand ils n'ont pas de message à envoyer ou recevoir. Nous pouvons distinguer trois classes de protocoles MAC : les protocoles MAC reposant sur TDMA, les protocoles à accès aléatoire

et les protocoles hybrides.

- Protocoles MAC reposant sur TDMA : Dans le protocole MAC basé sur l'accès multiple par répartition temporelle (*TDMA*) (Arisha *et al.*, 2002; Haartsen, 2000; Wu et Abouzeid, 2004a), chaque nœud a un intervalle de temps où il peut accéder au canal et utiliser toute la bande passante allouée par le capteur pour la transmission. *TRAMA* (Rajendran *et al.*, 2006) est parmi les premières techniques de gestion du trafic dans le réseau basée sur le protocole *TDMA*. *TRAMA*, en se basant sur un algorithme distribué, propose une attribution des intervalles de temps dynamique dans un rayon de deux sauts. Le temps d'accès au canal est divisé en deux périodes. Dans la première période d'accès, les nœuds échangent les adresses des voisins pour avoir une topologie de tous les voisins situés à deux sauts alors que dans la deuxième période le temps est divisé en plusieurs intervalles de temps.

*FLAMA* (FLow Aware Medium Access) (Rajendran *et al.*, 2005) est proposé comme une amélioration de *TRAMA*. Comme *TRAMA*, *FLAMA* divise le temps selon deux modes d'activité : intervalles de temps à accès planifié et intervalles de temps à accès aléatoire. En revanche, *FLAMA* échange durant les intervalles de temps à accès aléatoire des informations par rapport aux flux de données liés à l'application. *TDMA-ASAP* (Gobriel *et al.*, 2009) est un protocole qui intègre des algorithmes de coloriage qui donne au nœud le droit d'utiliser le temps d'accès du canal d'un nœud "frère" dans l'arbre. Ce protocole peut être adapté à plusieurs conditions de trafic réseau.

- Protocoles à accès aléatoire : L'accès aléatoire signifie que tous les nœuds partageant le canal de transmission ont le même droit d'accès à celui-ci. Cette solution permet une adaptabilité vis-à-vis des changements de densité et de topologie du réseau. Cependant, elle est également sujette aux conflits d'accès qui aboutissent à des collisions et donc à des pertes de paquets transmission et d'énergie au niveau de la couche MAC. *CSMA* et *CSMA/CA*.

L'accès multiple avec écoute de la porteuse (*CSMA* pour Carrier Sense Multiple Access) est introduit par (Kleinrock et Tobagi, 1975). Dans *CSMA* un nœud désirant transmettre un message, écoute le canal pour déterminer s'il est occupé. Dans ce cas, le nœud attend un temps aléatoire avant de tenter la retransmission. Quand le canal devient libre, le nœud transmet le message immédiatement. Une version plus étendue de *CSMA* avec évitement de collision *CSMA/CA* (*CSMA/CA* pour *CSMA* Collision Avoidance) a été proposé. Elle ajoute des mécanismes pour limiter le nombre de pertes de messages quand des nœuds voisins transmettent au même

moment. Les réseaux sans fil évitent les collisions en échangeant des messages de contrôle afin de réserver le canal avant chaque transmission de données.

(Ye *et al.*, 2002; Ye *et al.*, 2004) ont proposé le protocole Mac *S-MAC* (Sensor MAC). L'idée principale de ce protocole est de diviser le temps de fonctionnement du capteur en deux parties : active et endormi. Quand le capteur se met en actif, le capteur est prêt à transmettre ou à recevoir les données, tandis que dans l'état endormi, le capteur ne peut ni transmettre ni recevoir les données.

Cependant, il est impossible de changer le temps actif et endormi après le déploiement des capteurs ce qui empêche les capteurs de s'adapter aux différents niveaux de trafic. C'est pour cette raison que les auteurs dans (Van Dam et Langendoen, 2003) ont proposé *T-MAC* (Timeout- MAC). Dans *T-MAC*, les nœuds se synchronisent pour se mettre en mode actif et endormi périodiquement comme *S-MAC*. Cependant, l'intervalle de temps dans *T-MAC* n'est plus fixé par l'application mais varie selon le trafic du réseau. Si le trafic est important, les capteurs restent plus longtemps en mode actif afin de transmettre plus de données et si le trafic est faible, les capteurs restent un temps plus bref actifs afin d'économiser l'énergie.

Toujours basée sur la méthode de mise en mode actif et endormi périodiquement des capteurs, *D-MAC* (Lu *et al.*, 2004) offre une latence très faible si l'on compare avec d'autres protocoles MAC de la même famille. Le but de *D-MAC* est de que tout les nœuds inclus dans le chemin multi-saut vers le puits sont en mode actif quand les données sont en train d'être acheminées.

- Protocoles hybrides : Une troisième famille de protocoles propose de combiner les deux méthodes : TDMA et CSMA. Ainsi, ces protocoles essaient d'avoir les avantages des deux méthodes en alternant les deux dans le temps ou en les combinant d'une manière intelligente. *Z-MAC* (Zebra-MAC) (Rhee *et al.*, 2008) change dynamiquement de mode de transmission entre CSMA et TDMA en fonction de la charge actuelle du réseau. *Z-MAC* utilise CSMA comme protocole de base pour l'accès au support mais emploie un ordonnancement de type TDMA pour améliorer la résolution de contention entre les nœuds. Crankshaft (Halkes et Langendoen, 2007) est un protocole MAC hybride. Dans ce protocole, le temps est divisé en trames et chaque trame est composée de plusieurs slots. Pour recevoir des données, chaque nœud choisit un slot unicast ainsi que tous les slots de diffusion durant lesquels il écoute le canal. Ainsi, Crankshaft ordonnance la réception des données plutôt que leur envoi.

Funneling-MAC (Ahn *et al.*, 2006) est un protocole MAC hybride qui traite le

problème de congestion et de perte de paquets observé par les nœuds proches du puits. Le protocole utilise TDMA dans cette région là et CSMA ailleurs. TDMA est géré par le puits. Tous les nœuds utilisent CSMA par défaut, à moins de recevoir un message du puits les informant qu'ils doivent utiliser TDMA.

### 1.3.2 Les protocoles de routage efficaces en énergie

Ces protocoles de routage vont déterminer les chemins jusqu'au puits en considérant dans le calcul du coût d'une route des métriques relatives à la consommation d'énergie. La solution la plus triviale est de sélectionner le chemin ayant le plus petit nombre de sauts. Mais d'autres métriques sont importantes, en particulier le niveau de fiabilité des liens radio (les erreurs de transmission entraînent des retransmissions qui coûtent cher en énergie), le débit de transmission sur les liens radios et le niveau de charge des batteries des nœuds.

Il existe, beaucoup de propositions dans la littérature scientifique (Bhardwaj *et al.*, 2012; Jin *et al.*, 2012; Jones *et al.*, 2001; Al-Karaki et Kamal, 2004a). La majorité des propositions prend en compte la réserve d'énergie des nœuds, ceux dont le niveau de charge des batteries est le plus élevé sont privilégiés. Cette stratégie vise à prolonger la durée vie des nœuds (et donc celle du réseau) et à répartir plus efficacement la consommation d'énergie entre les nœuds du réseau.

Dans cette section, nous présentons une liste non exhaustive des protocoles de routages efficaces en énergie qui prennent en compte les contraintes imposées par les réseaux de capteurs sans fil (Akkaya et Younis, 2005).

#### Les Protocoles hiérarchiques

L'objectif principal du routage hiérarchique est de maintenir efficacement la consommation d'énergie des nœuds de capteurs en les impliquant dans la communication multi-sauts au sein d'un cluster et en effectuant l'agrégation et la fusion des données afin de diminuer le nombre de messages transmis au puits (Akkaya et Younis, 2005; Al-Karaki et Kamal, 2004b). La formation de clusters est généralement fondée sur la réserve d'énergie des capteurs et sur les capteurs qui sont à proximité de cluster-head. Deux grandes approches sont dérivées de ce type de protocole :

**Approche basée sur les groupes :** (Heinzelman *et al.*, 2000) proposent *LEACH*. L'idée de base de ce protocole est de former des clusters de nœuds de capteurs basés sur les zones, puis utiliser des chefs de cluster locaux comme passerelle pour atteindre

le puits. Cela permet d'économiser de l'énergie car les transmissions vers le puits ne sont effectuées que par le chef de cluster plutôt que par tous les nœuds de capteurs. *LEACH* choisit aléatoirement les nœuds *chef de cluster* et attribue ce rôle aux différents nœuds selon la politique de gestion tour à tour (Round-Robin) pour garantir une dissipation équitable d'énergie entre les nœuds. Cependant *LEACH* utilise un routage à saut unique ou chaque nœud peut transmettre directement au chef de cluster et au puits.

Une variante de *LEACH* appelée *LEACH-C* (Heinzelman *et al.*, 2002) a été conçue pour améliorer les performances de *LEACH*. Cette variante utilise une architecture centralisée pour choisir les chefs de cluster tout en impliquant le puits et l'information de localisation des capteurs. Cependant, elle augmente considérablement le trafic dans le réseau puisque tous les capteurs devront envoyer leurs informations de localisation au puits en même temps pendant chaque phase d'élection de chefs de cluster. Plusieurs travaux présentés (Perillo et Heinzelman, 2004) dans la littérature ont prouvé qu'une telle architecture est appropriée à des réseaux de petite taille et ne supporte pas le passage à l'échelle.

**Approche basée sur les chaînes (chaine-based approach) :** L'idée de formation de chaînes a été proposée pour la première fois dans l'algorithme *PEGASIS* (Power-Efficient Gathering in Sensor Information Systems) proposé par (Lindsey et Raghavendra, 2002) ; il est une version améliorée du protocole *LEACH*. Contrairement à *LEACH*, *PEGASIS* évite la formation des clusters et procure à un seul nœud dans la chaîne l'envoi de données au puits. Les données collectées sont transmises d'un nœud à un autre qui les agrège jusqu'à ce qu'elles arrivent à un nœud particulier qui les transmet au puits. Les nœuds qui transmettent les données au puits sont choisis selon un round-robin dans le but est de réduire l'énergie moyenne dépensée par un nœud.

En outre, *PEGASIS* suppose que tout nœud est capable de communiquer directement avec le puits. Or, cette supposition est loin de la réalité car les capteurs communiquent généralement en mode multi-sauts pour atteindre le puits.

*TEEN* (Threshold-sensitive Energy Efficient sensor Network protocol) proposé par (Manjeshwar et Agrawal, 2001), construit des groupements hiérarchiques à plusieurs niveaux où les nœuds plus proches forment des clusters jusqu'à ce que le puits soit atteint. Après la formation des clusters, chaque chef de cluster transmet à ses membres deux seuils : la valeur du paramètre à contrôler (seuil Hard) et une

petite variation de la valeur du paramètre contrôlé (seuil soft). Au début, les nœuds écoutent le médium en continu et lorsque la valeur captée du paramètre contrôlé dépasse le seuil Hard, le nœud transmet les données. La valeur captée est stockée dans une variable interne. Puis, les nœuds ne transmettront des données que si la valeur courante du paramètre contrôlé est supérieure au seuil hard ou diffère de la variable interne d'une quantité égale ou plus grande que la valeur du seuil Soft. Ce mécanisme réduit le nombre de messages à transmettre et par conséquent la consommation de l'énergie. Cependant, l'inconvénient principal de ce protocole est que, si les seuils hard et soft ne sont pas reçus, les nœuds ne communiqueront jamais, et aucune donnée ne sera transmise à l'utilisateur.

### Les Protocoles basés sur la localisation

Dans un protocole de routage basé sur la localisation, un nœud est supposé connaître sa position géographique, celle de ses voisins et celle du nœud destination. Ce type de protocole permet de construire le chemin jusqu'au puits à partir des informations sur les positions des nœuds. Dans (Xu *et al.*, 2001), les auteurs ont proposé un protocole de routage basé sur la localisation des nœuds nommé *GAF*. L'information de localisation utilisée dans *GAF* pourrait être fournie à l'aide d'un GPS ou d'autres techniques ou systèmes de localisation (Bulusu *et al.*, 2000; Doherty *et al.*, 2001). Le protocole de routage *GEAR*, suggéré par (Yu *et al.*, 2001), consiste à utiliser l'information géographique lors de la diffusion des requêtes aux régions cibles. L'idée est de restreindre le nombre de données dans la diffusion en prenant en considération uniquement une certaine région, plutôt que d'envoyer les données à l'ensemble du réseau.

### Les protocoles de routage "data-centric"

L'objectif du routage "Data-centric" est de conserver l'énergie en envoyant des requêtes à certaines régions et le puits attend à recevoir des données provenant des capteurs situés dans les régions sélectionnées. Comme les données sont demandées à travers des requêtes, le nommage est nécessaire pour préciser les propriétés des données (Subramanian, 2004).

Le routage data-centric concerne la méthode pour distribuer un type de données dans un réseau pour que les nœuds qui sont intéressés par ces données puissent les récupérer. Dans la littérature, il y a trois approches basées sur data-centric :

Le premier est SPIN (Sensor Protocols for Information via Negotiation) (Qi *et al.*, 2002) qui est conçu pour un réseau où un ou plusieurs nœuds veulent diffuser leurs données dans le réseau. Dans SPIN, les nœuds envoient seulement un paquet d'information qui

décrit les données. Quand un nœud reçoit ce paquet, si ce type de données l'intéresse, il va renvoyer une requête pour recevoir la donnée entière. SPIN est efficace dans la réduction des données redondantes et économise l'énergie (Akkaya et Younis, 2005).

Le deuxième est la diffusion dirigée (Intanagonwiwat *et al.*, 2003; Estrin *et al.*, 1999) : L'idée vise à diffuser des données aux nœuds en utilisant un schéma de nommage pour les données. La raison principale derrière l'utilisation d'un tel système est de se débarrasser des opérations inutiles de routage de couche réseau afin d'économiser l'énergie. Les nœuds intéressés par une donnée diffusent une requête. Les nœuds voisins prennent en compte cette requête, répondent par un message gradient et rediffusent à leur tour la requête.

Le troisième protocole est le protocole de routage par rumeur (Royer et Toh, 1999; Krishnamachari *et al.*, 2002) qui est principalement destiné pour des applications où le routage géographique n'est pas faisable. L'idée clé de cette méthode est de trouver les routes pour les requêtes vers les nœuds qui ont observé un événement particulier, plutôt que d'inonder tout le réseau pour récupérer des informations sur les événements survenus.

### 1.3.3 Contrôle de la topologie

Le contrôle de la topologie consiste à éliminer du réseau les nœuds inutiles (par la mise en sommeil des nœuds redondants) et les liens inutiles (par l'ajustement de la puissance de l'émetteur radio, et donc de la portée de la communication) pour diminuer la dépense d'énergie dans le réseau. Il s'agit donc de faire une réduction de la topologie initiale du réseau tout en préservant la couverture de la zone d'intérêt et la connectivité du réseau (Labrador et Wightman, 2009). Cette topologie réduite doit être maintenue à jour de temps en temps car le réseau a besoin d'évoluer au fur et à mesure que des nœuds actifs arrivent à épuisement.

Le contrôle de la puissance de l'émetteur radio n'a pas seulement un effet sur la durée de vie de la batterie des nœuds, mais aussi sur la capacité de charge du trafic. Le module de contrôle de la puissance est souvent intégré dans les protocoles soit de la couche réseau soit de la couche MAC.

Dans (Ingelrest *et al.*, 2006), les auteurs démontrent qu'il existe une portée de signal radio optimale qui minimise l'énergie dissipée tout en conservant la connectivité.

Quand le réseaux de capteurs est dense notamment, il est souhaitable de limiter le voisinage aux nœuds les plus proches pour réduire les collisions. Dans cette optique, plusieurs algorithmes de contrôle de topologie existent, en particulier *DRNG* et *DLMST* qui construisent la topologie réduite en se basant sur les informations collectées localement.

Avec *DLMST* (Li et Hou, 2004)(Directed Local Spanning Subgraph), chaque nœud connaît sa position et celle de son voisinage à un saut. Chaque nœud calcule un arbre couvrant minimum local parmi ses voisins directs. La construction de la topologie est basée sur l'arbre local de chaque nœud.

Avec *DRNG* (Li et Hou, 2004)(Directed Relative Neighborhood Graph), chaque capteur connaît sa position et la diffuse à son voisinage direct. Grâce à la position de son voisinage à 1 saut, un nœud va désélectionner les liens les plus longs.

### 1.3.4 Techniques de réduction des données

Les techniques de réduction des données dans les réseaux de capteurs sans fil visent à réduire la quantité des données à traiter et à transmettre. Ces techniques peuvent être classées, selon les étapes de traitement des données, en trois catégories : acquisition des données, traitement et transmission.

#### Acquisition des données

Certaines grandeurs physiques mesurées par les capteurs ne changent pas entre deux échantillons, c'est le cas par exemple d'une mesure de température où la dynamique est lente. Cela a poussé les chercheurs à exploiter la corrélation temporelle des données. Les techniques basées sur la collecte des données peuvent être classées en deux catégories : technique basée sur la prédiction et les techniques basées sur l'échantillonnage.

- Technique basée sur la prévision des données : Un modèle de prévision est établi lors des prélèvements de données, de telle sorte que les valeurs futures puissent être prévues avec une certaine précision. Cette approche exploite le modèle obtenu pour réduire le nombre de données acquies, et également la quantité de données à transmettre au puits.

Un exemple de la technique de prévision centralisée a été proposée dans (Goel et Imielinski, 2001). Les auteurs ont proposé un modèle de prédiction pour les applications de surveillance d'environnement nommé "*PREMON*" qui se base sur deux principes. Le premier est d'exploiter des corrélations spatiales. En connaissant les mesures d'un capteur, il devient inutile pour un voisin de retransmettre tout les paquets reçus mais seulement les variations pertinentes. Le deuxième se base sur le fait qu'un aperçu du réseau peut être considéré comme une image et par conséquent l'évolution des mesures peuvent être vu comme une vidéo. En explorant le concept de MPEG, *PREMON* génère un modèle de prévision dans le puits et l'en-



voie périodiquement aux noeuds capteurs. En effet, après déploiement, les noeuds capteurs envoient leurs mesures initiales au puits. Ensuite le puits calcule le modèle de prévision en exploitant les corrélations entre les données et l'envoie aux noeuds capteurs. Ainsi, dès réception du modèle, le noeud capteur compare ses données avec la valeur estimée du modèle de prévision. Si les deux valeurs sont proches, le noeud capteur ne transmet pas sa mesure au puits.

(Vuran et Akyildiz, 2006) a proposé une technique réduisant le nombre de transmissions. L'idée est de se servir de la corrélation spatiale parmi un ensemble dense de noeuds pour limiter le nombre de noeuds relayant l'évènement. Cela limite ainsi le nombre de collisions, améliore le délai d'acheminement et économise de l'énergie. Dans (Gedik *et al.*, 2007), un modèle de prédiction basé sur le regroupement nommé *ASAP* a été proposé. Les auteurs de *ASAP* proposent de regrouper les noeuds capteurs qui ont les mêmes mesures dans un même cluster. Le cluster-Head ainsi que le puits maintient le modèle de prévision. Et seulement les mesures dépassant un certain seuil sont transmis au puits.

Le protocole "buddy" présenté par (Goel *et al.*, 2006), se base sur le même principe que *PREMON* mais en utilisant un schéma distribué pour exploiter les corrélations entre les données. Dans le protocole "buddy", les noeuds capteurs sont regroupés pour former des clusters et dans chaque cluster, un seul noeud est choisi pour représenter son groupe.

- Technique basée sur l'échantillonnage : Les techniques basées sur l'échantillonnage peuvent être classées en trois catégories : L'échantillonnage adaptatif, l'échantillonnage hiérarchique et l'échantillonnage actif fondé sur un modèle.

Comme les échantillons mesurés peuvent être corrélés, les techniques d'échantillonnage adaptatif exploitent de telles similitudes pour réduire le nombre d'acquisition. L'approche d'échantillonnage hiérarchique suppose que les noeuds sont équipés de sondes (ou détecteurs) de différents types. Alors que chaque sonde est caractérisée par une résolution donnée et sa consommation d'énergie associée, cette technique choisit dynamiquement la classe à activer, afin d'obtenir un compromis entre la précision et l'économie d'énergie.

L'échantillonnage actif fondé sur un modèle adopte une approche semblable à la prévision de données. Un modèle du phénomène mesuré est établi lors des prélèvements de données, de telle sorte que les valeurs futures puissent être prévues avec une certaine précision. Cette approche exploite le modèle obtenu pour réduire le nombre d'échantillons de données, et également la quantité de données à transmettre au

puits, bien que ce ne soit pas leur objectif principal.

### Traitement et transmission des données

Plusieurs approches de traitement de données ont été proposées dans la littérature afin de faire face au problème de conservation d'énergie des capteurs sans fil. Nous présentons les deux principales. La première se base sur la compression des données et la deuxième sur l'agrégation des données (Marcelloni et Vecchio, 2008).

- Compression des données : Le paramètre critique des réseaux de capteurs sans fil est leur durée de vie. Or ce qui est coûteux en énergie ce sont les communications. L'intérêt des méthodes de compression est de permettre de communiquer moins de données, et donc d'économiser de l'énergie, ce qui se traduit par un allongement de la durée de vie des réseaux. Cependant, les algorithmes de compression utilisés couramment sur ordinateur (comme Lempel-Ziv, WinZip ou JPEG) ne sont tous directement transférables aux capteurs, car beaucoup supposent trop de calculs en comparaison du gain apporté.

Il existe plusieurs méthodes de compression des données. Les principales méthodes de compression des données adaptées aux réseaux de capteurs sans fil sont :

- Le codage par ordonnancement : La compression dans un codage par ordonnancement consiste à supprimer les informations redondantes des capteurs (l'adresse destination, codes de contrôle des erreurs, synchronisation de l'horloge) et fusionner les données restantes. Ainsi, le nombre de paquets envoyés est réduit. Dans le codage par ordonnancement, les nœuds envoient leurs paquets à un nœud unique appelé "compression node". Le paquet codé est construit à partir de combinaisons des paquets reçues à partir de plusieurs nœuds. Le protocole "Funning" (Petrovic *et al.*, 2003), basé sur l'approche par ordonnancement, réduit considérablement la quantité d'énergie consommée lors de la communication qui peut atteindre 44% d'économie d'énergie. Cet objectif peut être réalisé en n'envoyant qu'un seul flux de données pour un groupe de capteurs au lieu que chaque capteur envoie individuellement ses données.
- La compression distribuée (Pradhan *et al.*, 2002) : Chaque capteur fournit une représentation binaire de ses échantillons en prenant en compte la corrélation qui existe avec les échantillons mesurés par les autres capteurs. La communication entre capteurs est interdite ce qui rend l'estimation de cette corrélation difficile en pratique.
- Agrégation des données : Les réseaux de capteurs étant assez denses en général, cela

signifie que des nœuds assez proches (voisins) peuvent capter les mêmes données (température, pression, humidité équivalentes par exemple). Par conséquent, plusieurs études ont été menées pour éliminer la redondance et réduire le trafic des données dans le réseau. Le mécanisme consiste à traiter les données recueillies par chaque capteur au niveau d'un nœud appelé "nœud agrégateur". Seulement le résultat produit sera transmis à la station de base. De cette manière, la quantité de données communiquées dans le réseau peut être diminuée, ce qui réduit par conséquent la consommation de la bande passante et l'épuisement d'énergie des capteurs.

Les techniques d'agrégation de données peuvent être découpées en deux :

- Agrégation centralisée : Agrégation dans des clusters, formés via un protocole de clustering. On définit d'abord des zones (clusters) via un protocole puis ensuite on agrège les données dans ces zones grâce à un cluster-head. Ce dernier peut éventuellement changer au cours du temps afin de répartir au mieux la consommation d'énergie entre tous les nœuds du réseau.
- Agrégation distribuée : L'agrégation distribuée est une agrégation dans un arbre c'est-à-dire que le réseau est vu de façon globale. Le protocole COUGAR (Yao et Gehrke, 2002) est un exemple d'agrégation de données distribuée. Les données produites par le réseau sont modélisées comme une table relationnelle, le réseau étant vu comme une grande base de données distribuée. Les attributs de cette table sont soit des informations sur le capteur soit des données produites par ce capteur car COUGAR fournit une agrégation partielle au niveau des nœuds.

## 1.4 Le cas des réseaux de capteurs d'images

Si le paradigme des réseaux de capteurs sans fil a été défini il y a déjà une quinzaine d'années (le lancement du projet Smart Dust (Pister, 1997) par Kris Pister, Joe Kahn et Bernhard Boser de l'université de Californie-Berkeley en 1997 constitue le premier fait marquant), l'engouement pour les réseaux de capteurs d'images est plus récent. Il coïncide avec la commercialisation de mini-caméras en technologie CMOS, consommant très peu d'énergie, comme par exemple la caméra *ADCM-1650* chez Agilent Technologies. Une communauté scientifique s'est formée sur le sujet à partir de 2005 sous l'impulsion de Center for Embedded Networked Sensing (CENS, UCLA ), concepteur du fameux capteur d'image Cyclops (Rahimi *et al.*, 2005).

Comparés aux réseaux de capteurs classiques qui manipulent des données scalaires

(mesure de température, d'humidité, etc), les réseaux de capteurs d'images ont quelques différences très importantes :

- Le volume d'information associé à une image est de plusieurs ordres de grandeurs supérieur à une information scalaire simple, et cela change tout. En effet, les données fournies par un capteur traditionnel sont codées généralement sur quelques bits (sur la carte d'acquisition MTS 400 de Crossbow par exemple, les valeurs de température et l'humidité sont fournies sur 14 bits par le convertisseur analogique/numérique, les valeurs de luminosité sur 12 bits). Par conséquent, elles peuvent être transportées sur un seul paquet et la compression des données n'a donc pas vraiment d'intérêt. Une image est au contraire représentée sur plusieurs milliers d'octets et donc son transport nécessite typiquement plus d'un paquet. La compression des données d'image est une nécessité pour réduire le nombre de paquets à émettre, d'autant qu'en pratique, le nombre de paquets générés par une image est très grand car la contrainte énergétique propre aux réseaux de capteurs impose de construire de paquets de petite taille. La raison vient du fait que transmettre des paquets de grande taille, bien que cela réduit la surcharge de trafic relative aux entêtes des protocoles, augmente la probabilité d'erreurs de transmission dues aux imperfections de la liaison radio. La question de la taille optimale des paquets sous la contrainte de l'efficacité énergétique des réseaux multi-sauts a été étudiée dans (Sankarasubramaniam *et al.*, 2003). Sur la base de ces travaux, la taille maximale des paquets définie par défaut dans TinyOS est de 36 octets seulement, dont 29 pour la charge utile. Dans cette configuration, il faut plus de 600 paquets pour transmettre une image monochrome de taille  $128 \times 128$  (8bpp) si elle n'est pas compressée à la source, et environ 40 paquets si l'image est compressée à 0.5 bpp.
- Les images naturelles possèdent une certaine tolérance aux pertes de pertes de paquet puisqu'il y a des corrélations spatiales entre les pixels voisins. Cela veut dire que le protocole de communication n'a pas besoin de fournir un service de transport totalement fiable, et des économies d'énergie peuvent être trouvées à ce niveau. Précisons toutefois que la tolérance aux pertes de paquets diminue si l'image est comprimée avant transmission puisque l'opération de compression vise à éliminer les corrélations spatiales entre les pixels.
- La plupart des capteurs traditionnels ont un champ de perception omnidirectionnelle, c'est-à-dire que l'orientation du capteur n'a pas d'impact sur la valeur de la grandeur physique mesurée. Des nœuds très proches les uns des autres peuvent alors être considérés comme redondants puisque les mesures qu'ils fournissent sont les

mêmes. Le champ de vision des caméras est par contre restreint en direction et donc l'hypothèse que des capteurs voisins sont implicitement redondants ne tient plus. La réduction de la consommation d'énergie par contrôle de la topologie (réduction de la topologie par mise en sommeil des nœuds redondants) est donc plus complexe à mettre en œuvre dans les réseaux de capteurs d'image.

Parmi ces trois différences caractéristiques des réseaux de capteurs d'image, nous nous sommes intéressés essentiellement à la compression des données pour contribuer au problème de la consommation d'énergie. L'état de l'art que nous allons présenter porte aussi sur les travaux liés à la transmission d'image car compression et transmission sont étroitement liés. Avant cela, nous présenterons quelques exemples d'application.

### 1.4.1 Exemples d'application

Les réseaux de capteurs d'image concernent un vaste champ d'applications, en fait toutes celles qui touchent à la détection, la reconnaissance, la localisation et le dénombrement d'objets par la vision. Les applications militaires (détection d'intrusions, espionnage de l'ennemi) sont les premières concernées.

Un réseau de capteurs expérimental a par exemple été déployé en 2003 par une équipe de l'Université de l'état d'Ohio sur la base *MacDill* de l'US air force à Tampa, Floride (Arora *et al.*, 2004). Il était conçu pour détecter et classifier trois types de cibles : des personnes non armées, des soldats et des véhicules. Le réseau était constitué de 90 nœuds équipés d'un capteur magnétique (pour la détection des métaux), d'un radar Doppler et d'une caméra.

Les réseaux de capteurs d'image sont aussi très utiles pour l'observation de la faune et de la flore en milieu naturel. Le projet *PODS* (Biagioni et Bridges, 2002) développé par l'université de Hawaï avait pour objectif d'identifier les raisons pour lesquelles certaines plantes menacées poussent dans certaines régions mais pas dans d'autres. Les capteurs d'images nommés *PODS* sont déployés dans le parc national des volcans d'Hawaï. *PODS* collecte la température chaque 10 minutes et les images sont collectées chaque heure et ensuite transmises à l'utilisateur final.

Le projet *BearCam* (Wawerla *et al.*, 2009) est un autre exemple d'application pour l'observation de la nature. Le but de ce projet était de surveiller les ours bruns à la sortie de leur sommeil hivernal. Le système aide les biologistes à collecter des données pour leurs études de l'effet de la présence de l'être humain sur le comporte-

ment des ours. De même dans (Savvides et Srivastava, 2002), un réseau de capteurs pour surveiller la reproduction des oiseaux a été réalisé. Un système composé d'une vingtaine de capteurs d'image Cyclops a été déployé dans la réserve des Montagnes James San Jacinto (Californie).

Des capacités de vision sont aussi nécessaires dans les applications de surveillance d'objets mobiles pour les différencier par leur forme (Chang et Aghajan, 2006), leur couleur (Oto *et al.*, 2006) et faciliter leur repérage géographique (Lee et Aghajan, 2006).

### 1.4.2 Travaux relatifs aux réseaux de capteurs d'images

Les travaux issus de l'état de l'art dans le domaine des réseaux de capteurs d'image sous la contrainte de la limitation des ressources peuvent être classés en trois catégories : ceux traitant de la compression de l'image à la source, ceux portant sur la compression distribuée et ceux portant sur la transmission d'image compressée ou non.

#### Etat de l'art sur la compression d'image à la source

Ici la compression de l'image est réalisée par un algorithme local, c'est-à-dire que tout le traitement est supporté par le nœud caméra initiateur de la prise d'image. Cette partie de l'état de l'art est la plus abondante car la compression d'image à la source est ce qu'il y a de plus facile à mettre en œuvre. La plupart des travaux proposent des modifications des algorithmes bien connus JPEG (Taylor, 2004; Mammari *et al.*, 2008), JPEG2000 (Lian *et al.*, 2003; Chiang *et al.*, 2004) ou SPIHT (Kim et Ra, 2004) pour réduire leur coût de calcul et/ou de mémoire. Tous ces algorithmes standards s'appuient sur une étape de décorrélation des données basée sur une transformée (La DCT pour JPEG, la DWT pour JPEG2000 et SPIHT), suivie d'une étape de quantification puis un codage entropique des coefficients.

Dans le cas de JPEG, où l'image est divisée en bloc de  $8 \times 8$  pixels, (Taylor et Dey, 2001; Mammari *et al.*, 2008) réduisent le coût de la DCT en ne calculant qu'une partie des coefficients parmi les 64 initiaux. Un masque de forme carrée était adapté par (Taylor et Dey, 2001) alors que (Mammari *et al.*, 2008) adoptait un masque de forme triangulaire.

Un autre moyen de réduire le coût des étapes de DCT et de quantification est de faire les opérations sur des réels en virgule fixe plutôt qu'en virgule flottante. (Chiasserini et Magli, 2002) ont obtenu une baisse de la consommation d'énergie d'un facteur de 8 environ sur un microcontrôleur *StrongArm SA1110*.

(Lee *et al.*, 2007) ont décrit une méthode pour déterminer le nombre de bits optimal pour la partie entière et la partie fractionnelle des réels en virgule fixe en fonction de la précision recherchée et la taille du bus de données du microcontrôleur.

Les algorithmes de compression basés sur la DWT sont aussi coûteux en temps et en énergie. (Ferrigno *et al.*, 2005) ont par exemple rapporté que le temps d'exécution de SPIHT et de JPEG2000 était respectivement 2 et 6 fois plus long que JPEG. Le cas de JPEG2000 est particulièrement significatif. (Moccagatta et Coban, 2000) ont évalué que la complexité de l'encodeur JPEG2000 était entre 38% et 264% supérieur à JPEG. Un document d'information des membres de comité JPEG (Santa-Cruz et Ebrahimi, 2000) présente des résultats similaires. La majorité du temps d'exécution vient du codage EBCOT qui requière de nombreuses opérations sur bit qui ne peuvent être exécutées efficacement en implantation logicielle. Deux méthodes sont présentées dans (Lian *et al.*, 2003; Chiang *et al.*, 2004) pour réduire la complexité d'EBCOT et donc améliorer le temps d'exécution de JPEG2000. Un autre problème lié à l'utilisateur de la DWT est son coût mémoire puisqu'elle s'applique typiquement sur l'image entière contrairement à la DCT qui s'applique sur des blocs de petite taille. Par conséquent la mémoire interne à réserver pour stocker les résultats intermédiaires est de grande dimension. Pour faire face à ce problème, (Rein et Reisslein, 2011) ont proposé une méthode pour calculer les coefficients de la DWT avec très peu de mémoire, typiquement pour une image de taille  $N \times N$  un buffer de  $9 \times N$  octets si les coefficients sont représentés en virgule flottante, et de  $5 \times N$  octets seulement s'ils sont représentés en virgule fixe, au prix toutefois d'un nombre d'additions et de multiplications nettement plus élevé.

Quelques travaux (Hicks *et al.*, 2008; Duran-Faundez *et al.*, 2011) s'affranchissent des algorithmes de compression standards et adoptent un schéma ne faisant appel à aucune transformée. Ils offrent ainsi des coûts de calcul et des coûts mémoires nettement plus petits, même si, en contrepartie, ils ne peuvent pas atteindre des taux de compression aussi élevés. (Hicks *et al.*, 2008) ont proposé une variante de Packbits, originellement un système de compression sans perte développé par *Apple* utilisant simplement des codes de longueurs variable. Cette variante permet de faire de la compression avec perte et donc d'obtenir de meilleurs taux de compression. Elle utilise une technique de seuil pour représenter des valeurs consécutives proches les unes des autres avec le même code. Les auteurs rapportent qu'avec un seuil fixé à 5, la qualité visuelle de l'image reste très bonne tout en permettant de réduire la taille des données d'environ 25%. Lorsque le seuil est fixé à 0, cela revient à utiliser Packbits dans sa version sans perte. L'inconvénient de ce système de compression est qu'il exige d'utiliser un protocole de communication fiable, le

décodeur ne pouvant pas travailler correctement s'il manque des données.

(Duran-Faundez *et al.*, 2011) ont présenté aussi un algorithme de compression sans transformée, nommé TiBS (Tiny Block-Size Coding), qui traite l'image par bloc de  $2 \times 2$  pixels. Il fournit des taux de compression nettement meilleurs que Packbits, le volume de données pouvant être divisé entre 2 et 5 selon la qualité d'image recherchée. Surtout, en encodant l'image par blocs de taille minuscule, TiBS permet de renforcer la tolérance aux pertes de données puisqu'un bloc manquant au décodeur sera d'autant plus facile à masquer qu'il est petit. Un protocole non fiable peut être adopté pour transporter les paquets de données, d'où un coût d'énergie très faible pour l'ensemble des nœuds de réseau.

### Etat de l'art sur la compression d'image distribuée

Il ne faut pas confondre la compression d'image distribuée, au sens où le traitement d'une image sera réparti sur plusieurs nœuds de manière coopérative, et la compression de source distribuée (DSC), où des images prises par des caméras distinctes mais mutuellement corrélées sont compressées séparément et décodées conjointement (selon des principes basés sur les théorèmes de *Slepian-Wolf* pour la compression à perte et de *Wyner-Ziv* pour la compression à perte). En d'autres mots, il n'y a pas de communication entre les sources et chacune encode ses données indépendamment des autres. Une caractéristique principale de la DSC est que les nœuds sources ont des encodeurs de très faible complexité, la complexité se situant au niveau du décodeur étant donc supportée par le puits. La DSC a été étudiée dans le contexte des réseaux de capteurs et récemment, quelques travaux se sont intéressés au codage d'images et de séquences vidéo (Wagner *et al.*, 2003; Thirumalai *et al.*, 2008).

Cependant, comme dit en (Soro et Heinzelman, 2009), plusieurs verrous doivent être levés avant que la DSC puisse être mise en pratique dans les réseaux de capteurs d'image. Par exemple, il est extrêmement difficile de définir la structure des corrélations entre images différentes, particulièrement quand la topologie du réseau est inconnue ou sans phase d'entraînement.

Développer un système de coopération entre les nœuds du réseau pour répartir le coût de la compression d'image semble finalement approprié au problème de la contrainte des ressources dans les réseaux de capteurs. Il s'agit pour le nœud source de découper l'image en plusieurs morceaux et de les envoyer à des nœuds distincts pour qu'ils exécutent, chacun, une partie du traitement.

Dans (Wu et Abouzeid, 2004b), un système de coopération est proposé pour distribuer



le coût de calcul de DWT dans le cadre d'un codage JPEG2000.

Ce système de coopération, qui s'appuie sur une architecture clustérisée, est donné en deux variantes : une méthode de DWT parallèle et une méthode par carrelage.

Dans la première méthode, l'application d'une transformée d'ondelettes parallèle (Marino *et al.*, 1999) est proposée. L'image capturée est divisée en  $n$  blocs de données  $R_1, \dots, R_n$ , consistant en une ou plusieurs lignes. Ces blocs sont transmis à certains noeuds voisins de la source. Ces noeuds effectuent une transformée en ondelettes unidimensionnelle (1-D) pour chaque bloc de données qui lui a été envoyé, puis transmet le résultat à un noeud agrégat. Celui-ci divise les données obtenues, cette fois en  $m$  blocs  $I_1, \dots, I_m$  composé de colonnes. Puis il distribue ces blocs en les transmettant à des noeuds voisins. Ceux-ci effectuent alors la transformée en ondelettes 1-D et les renvoient au noeud agrégat. Celui-ci récupère tous les blocs et obtient aussi le résultat de transformation bidimensionnelle (2-D) en ondelettes de l'image.

Dans la deuxième variante, l'image capturée est divisée en tuiles (de blocs de données), puis ces tuiles sont distribuées à un certain nombre de noeuds voisins. Ces noeuds effectuent la transformation en ondelettes 2-D de façon indépendante sur ces tuiles. Enfin, les résultats sont transmis à un noeud agrégat.

Ce système de coopération n'a pas pour objectif de minimiser la dépense d'énergie de l'ensemble du réseau, mais plutôt de la répartir plus équitablement entre les noeuds afin de prolonger la durée de vie du réseau. Les résultats de simulation donnés dans (Wu et Abouzeid, 2004b) montrent que le réseau a une durée de vie quatre fois plus grande avec ce système de coopération comparé au cas où la compression de l'image est exécutée localement à la source.

Dans (Lu *et al.*, 2008), le même système est repris mais la compression d'image est basée sur la transformée orthogonale à recouvrement (LBT ou Lapped Biorthogonal Transform), un codage par arbre de zéro et un codeur arithmétique. Ce schéma de compression exige moins de calculs que JPEG2000, et par conséquent est moins gourmand en énergie.

Finalement, la compression distribuée d'image de manière coopérative offre une solution élégante à la limitation des capacités de calcul et de mémoire des noeuds capteurs. Toutefois, la négociation et l'ajustement dynamique des coopérations entre les noeuds passe par un supplément de communication entre eux et typiquement, ces échanges doivent être mis en œuvre par un protocole de communication fiable. Son coût en énergie doit donc être évalué, notamment dans des conditions où les liaisons radio sont sujettes à des erreurs de transmission.

## Etat de l'art sur la transmission d'images

Malgré les imperfections potentielles des liaisons radio, l'image reçue au puits doit avoir une qualité visuelle satisfaisante pour l'utilisateur final. Le problème posé sur le protocole de communication est de fournir un niveau de fiabilité suffisant pour que la contrainte de la qualité de l'image soit satisfaite tout en limitant la dépense d'énergie.

(Chiasserini et Magli, 2004) ont comparé la dépense d'énergie pour transmettre des images compressées avec JPEG dans le cas où le protocole de communication utilise un schéma ARQ et dans le cas où il utilise un schéma FEC. Leurs résultats analytiques montrent que le schéma ARQ est toujours moins coûteux en énergie que le schéma FEC mais que si les délais de bout en bout sont à prendre en compte dans la performance (cas de flux vidéo par exemple), alors le schéma FEC est justifié lorsque le taux d'erreurs sur les canaux de transmission est élevé. (Tian *et al.*, 2008) ont aussi étudié le coût énergétique des schémas ARQ et FEC en considérant la taille des paquets et la distance entre les nœuds. Ils considéraient les caractéristiques matérielles des nœuds Mica2. Ils montrent analytiquement que ARQ est avantageux lorsque la taille des paquets est grande et que la distance entre les nœuds est petite, et vice versa.

Une approche pour réduire la consommation d'énergie liée à la communication est de préparer à la source des paquets de différentes priorités en décomposant l'image en plusieurs niveaux de résolution sur la base d'une DWT. De cette manière, le protocole de communication peut engager un contrôle d'erreur différencié et accepter des pertes pour les paquets de faible priorité. Les paquets de la plus haute priorité sont transmis fidèlement et donc, une qualité minimale de l'image reconstruite est toujours garantie. Cette approche a été suivie dans (Lecuire *et al.*, 2008). Ils proposent un protocole de communication qui conditionne l'acheminement des paquets, à chaque saut, à l'état de charge de la batterie du nœud de transit et à la priorité des données. Il est basé sur un modèle de commande dynamique qui tient compte de la proximité du nœud avec le puits.

Une approche de routage multi-chemin est aussi intéressante dans le transport d'image, soit pour envoyer sur chaque chemin une copie des paquets pour augmenter sa probabilité d'arrivée au puits, soit pour répartir le trafic, et donc la dépense d'énergie sur un plus grand nombre de nœuds (Maimour, 2007).

Enfin, une dernière approche consiste à adopter un protocole de communication non fiable, c'est-à-dire d'accepter la perte de n'importe quel paquet. Cette approche n'est envisageable que si le système de compression d'image à la source a une grande tolérance aux pertes de paquets. (Duran-Faundez et Lecuire, 2008) proposent de combiner un système

de compression par blocs comme JPEG ou TiBS avec un algorithme de mélange pour prendre soin d'embarquer dans chaque paquet des blocs d'images non contigus. Cela permet de renforcer la tolérance du système de compression aux pertes de paquets puisqu'un bloc manquant au décodeur sera plus facile à masquer si les blocs voisins ont été reçus. Ils présentent un algorithme de mélange chaotique qui a un coût de calcul et de mémoire négligeable, et donc qui est bien adapté au problème de la limitation des ressources des nœuds capteur.

## 1.5 Conclusion et Perspectives

Les besoins d'applications pour les réseaux de capteurs d'images deviennent de plus en plus nombreux. La demande aujourd'hui concerne notamment les applications environnementales ou le pilotage de robots. Avec la contrainte de la limitation des ressources des nœuds, des mécanismes de traitement et de transmission d'images efficaces en énergie restent à étudier. Dans ce chapitre, nous avons présenté les travaux trouvés dans la littérature, qui traitent essentiellement le problème de l'énergie dans les réseaux de capteurs sans fil. Nous avons présenté un cas spécifique de réseau de capteurs sans fil : les réseaux de capteurs d'images. Comparés aux réseaux de capteurs classiques qui manipulent des données scalaires, les réseaux de capteurs d'images traitent un volume d'informations beaucoup plus grand. Nous nous sommes intéressés à la compression d'images pour diminuer le volumes des données à transmettre afin réduire la consommation de l'énergie.

Dans ce domaine, on remarque des efforts de plusieurs auteurs pour adapter les algorithmes de compression d'images, comme JPEG ou JPEG2000, aux contraintes particulières des réseaux de capteurs sans fil. Nos contributions qui seront développées dans les chapitres suivants, suivent aussi cette voie en considérant le cas de JPEG. Nos propositions seront validées par expérimentations sur de vrais capteurs.



# Chapitre 2

## Compression d'images et contributions

### Sommaire

---

<b>2.1</b>	<b>Généralités</b>	<b>36</b>
<b>2.2</b>	<b>Les techniques de base de la compression d'images</b>	<b>38</b>
2.2.1	Techniques de décorrélation intra-image	38
2.2.2	Quantification	44
2.2.3	Codage entropique	47
2.2.4	Synthèse	51
<b>2.3</b>	<b>Réduction DCT</b>	<b>51</b>
2.3.1	L'approche par factorisation de la matrice (DCT rapide)	53
2.3.2	L'approche zonale (DCT réduite)	58
2.3.3	La DCT zonale rapide	60
2.3.4	Evaluation des performances	73
<b>2.4</b>	<b>Conclusion</b>	<b>76</b>

---

Les images naturelles représentent de grands volumes de données, la compression d'image est donc vitale dès lors que l'on veut les transmettre sur les réseaux de capteurs d'images sans fil. Ce chapitre a pour but de synthétiser ce domaine spécifique du traitement d'images en présentant les approches principales de compression et les problèmes élémentaires liés à une implantation sur le nœud caméra, ce qui permettra d'aborder les questions les plus importantes comme la consommation d'énergie des nœuds-capteurs.

Notre contribution porte sur la proposition d'intégrer dans une chaîne de compression de type JPEG une DCT zonale rapide, autrement dit de combiner une méthode de sélection zonale des coefficients avec une méthode de DCT rapide. Cela réduit 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.

Après avoir présenté les principes de la compression d'images, nous introduisons un modèle qui explicite la réduction du coût de calcul de la compression. Ce modèle sert de base à l'étude et au développement scientifique de cette thèse. Il pilote le bilan de l'état de l'art des méthodes de calcul de la DCT rapides dans la section 2.3.2, de quantification et de codage suivi par une évaluation des performances des différentes étapes de processus de la compression d'images. Cela nous permet de donner certaines justifications à l'utilisation de l'algorithme de calcul de la DCT rapide basé sur l'algorithme *Cordic Loeffler*, de la quantification *scalaire* et du codage de *Golomb et arithmétique* dans la compression des images.

Etant donné que nous examinons le sujet de la compression dans le cadre de réseaux de capteurs, à la fin de chapitre, nous présentons un modèle de compression avec une complexité de calcul réduite.

## 2.1 Généralités

Une image contenant des milliers de pixels, sa gestion consiste à traiter -pour stocker et/ou transmettre- cette quantité importante de données. Une seule image à niveau de gris (de taille standard  $512 \times 512$ ) codée sur 8 bits par pixel représente 256 Ko sous forme brute, non compressée, cela devenant trois fois plus important dans le cas d'une image en couleur.

L'intérêt de la compression est donc évident. Les images compressées prennent moins de temps de transmission sous forme compressée sur le même canal, ou bien, elles ont besoin d'une bande passante plus petite pour arriver à destination en même temps que la même image non compressée.

On peut distinguer deux grandes familles d'algorithmes de compression : les méthodes dites sans perte ou réversibles garantissent la restitution parfaite des images, alors que les méthodes dites avec perte ou irréversibles modifient plus ou moins la valeur des pixels.

La compression sans perte permet d'exploiter les redondances du signal mais ne permet pas des taux de compression élevés. Comme l'utilisateur final des images ne s'intéresse souvent qu'à l'information pertinente de l'image, on préfère la compression avec pertes puisque avec ce type de compression, on peut atteindre des taux de compression plus importants en fonction du niveau de qualité de l'information accepté par rapport à l'image initiale et du type de compression adopté.

**Le but de la compression d'image dans le contexte des réseaux de capteurs sans fil est donc de modifier la représentation initiale des données, pour qu'on**

ait moins de paquets à envoyer via le réseau tout en augmentant la durée de vie du nœud caméra. Cette nouvelle représentation sera décodée durant une procédure de décompression pour reconstruire l'image.

Le taux de compression donne une mesure de performance des méthodes de compression des images fixes :

$$\text{taux} = \frac{\text{nombre de bits dans l'image originale}}{\text{nombre de bits dans l'image compressée}} \quad (2.1)$$

Cependant, le taux de compression n'est qu'un seul critère d'évaluation des algorithmes de compression. Tout d'abord, dans le cas des approches irréversibles, le taux de compression ne peut évaluer la performance de la compression qu'avec la distorsion introduite (mathématiquement calculée ou/et visuelle). Les autres critères qui sont à considérer sont la complexité, l'énergie consommée lors du codage, la robustesse aux erreurs de transmission ou la flexibilité aux besoins de l'utilisateur en particulier dans le cas des réseaux de capteurs sans fil.

Comme nous le laissons sous-entendre, parmi les deux possibilités, nous avons choisi dans cette thèse d'examiner et d'appliquer les approches avec pertes. En effet, la performance en termes de taux de compression des algorithmes réversibles est très limitée.

Les approches irréversibles sont basées sur les redondances présentes dans l'image. En même temps, elles exploitent le fait que l'oeil n'est pas sensible à certaines dégradations. Par exemple, le nombre de niveaux de gris que l'oeil peut distinguer est largement inférieur à 256, il n'est même pas de 64. La quantification devient ainsi une technique de base pour les méthodes irréversibles.

Ce processus de quantification peut être plus ou moins complexe, par exemple en s'appliquant sélectivement sur le signal de manière à maximiser la qualité de reconstruction pour un coût donné. Classiquement, en compression d'image on se base sur l'erreur quadratique moyenne (mean square error, MSE) de chaque image :

$$MSE = \frac{1}{N \times M} \sum_{i=1}^M \sum_{j=1}^N |x_{rec}[i, j] - x[i, j]|^2 \quad (2.2)$$

Dans la littérature, les mesures de qualité sont en général exprimées en rapport signal sur bruit, soit, dans le cas d'un signal représenté sur 8 bits :

$$PSNR = 10 \log_{10} \left( \frac{255^2}{MSE} \right) \quad (2.3)$$

## 2.2 Les techniques de base de la compression d'images

La problématique de la compression peut être considérée de différents points de vue. Dans cette section, nous cherchons à établir un concept qui guidera l'étude des méthodes de compression qui répondent aux contraintes des réseaux de capteurs sans fil. Nous allons développer un schéma dont les éléments sont couramment utilisés dans la communauté scientifique de la compression d'image, mais, à notre connaissance, qui n'ont jamais été rassemblés pour former un modèle pratique et optimal pour les réseaux de capteurs sans fil.

Le schéma de principe d'un encodeur d'image souvent utilisé pour décrire le fonctionnement des algorithmes de compression est celui présenté dans la FIGURE 2.1.

La première étape est une transformation, ayant pour but de réaliser un changement de l'espace de représentation du domaine spatial. L'étape de quantification réalise une réduction de nombre de valeurs représentées, et fait partie uniquement des algorithmes irréversibles. Le codage permet de passer d'une représentation de données vers une autre sans perte sur les valeurs quantifiées.



FIGURE 2.1: Schéma général de compression avec perte

L'évaluation que nous allons proposer dans les sections suivantes porte sur différentes étapes de la chaîne de compression. L'objectif est de fournir une chaîne de compression adaptée aux contraintes des réseaux de capteur (faible coût calculatoire, faible coût de stockage... ). Pour que la compression de données soit rentable en terme d'économie d'énergie, il faut que le traitement et la transmission des données compressées coûte moins d'énergie que la transmission d'une image non compressée.

### 2.2.1 Techniques de décorrélacion intra-image

Le principe de base consiste à éliminer l'information qui n'affecte pas ou peu l'aspect visuel. Nous présentons dans cette section les méthodes par transformation les plus connues.

Les pixels composant les images sont en général corrélés et possèdent donc une information redondante. Une étape de décorrélacion visant à supprimer cette redondance s'avère



alors très bénéfique pour l'amélioration des taux de compression. L'approche principale qui sert à effectuer cette décorrélation est la transformation.

Les techniques intégrant une étape de transformée figurent parmi les techniques de compression les plus employées. Elles permettent d'obtenir des taux de compression élevés tout en conservant une bonne qualité d'image. Elles sont utilisées par des standards internationaux pour le codage des images fixes comme JPEG et JPEG200. Cette étape de traitement, réversible, produisant des coefficients décorrés et concentrant l'énergie sur peu de coefficients, permet d'optimiser les étapes ultérieures de quantification et de codage.

Dans cette section, nous comparons les deux approches les plus utilisées à savoir la transformée en ondelettes (Mallat, 1999) et la transformée en cosinus discrète (Ahmed *et al.*, 1974).

### La transformée en cosinus discrète (DCT)

La transformée en cosinus discrète est une transformation mathématique complexe qui a pour but de transformer le domaine de représentation de l'image. Elle permet de passer d'un domaine spatial à un domaine fréquentiel. Notre œil est moins sensible à certaines données de l'image, le but de la DCT est de passer dans un domaine fréquentiel, ce qui permettra de trier efficacement l'ensemble des données de l'image et ainsi supprimer certaines données peu sensibles à l'œil humain, cela revient à supprimer les hautes fréquences de l'image tout en gardant les données majeures qui sont représentées par les basses fréquences.

Cependant, la DCT est particulièrement adaptée pour décrire des signaux stationnaires, ce qui n'est pas le cas pour les images naturelles. Pour pallier cette limitation, l'image est décomposée en blocs avant de transformer chaque bloc indépendamment.

Durant la dernière décennie, la DCT est apparue comme transformation d'image dans la plupart des systèmes visuels. Comme d'autres transformations, la DCT tente de décorréler les données de l'image.

À partir des intensités de blocs de  $N \times N$  pixels  $I(i, j)$ , nous obtenons les coefficients DCT bidimensionnelles (2-D DCT) associés  $F(u, v)$ . La définition de la 2-D DCT est donnée dans l'équation 2.4.

$$F(u, v) = \frac{2}{N} C(u)C(v) \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} I(i, j) \cos\left(\frac{\pi(2i+1)u}{2N}\right) \cos\left(\frac{\pi(2j+1)v}{2N}\right) \quad (2.4)$$

Où :

$$C(x) = \begin{cases} 1, & \text{si } x \neq 0 \\ \frac{1}{\sqrt{2}}, & \text{si } x = 0 \end{cases}$$

Comme on l'a déjà dit, la DCT est particulièrement adaptée à des signaux stationnaires. Cette hypothèse n'étant pas respectée (les images naturelles sont non stationnaires), on effectue la transformée sur de petites portions de l'image, généralement de taille  $8 \times 8$ , pour en atténuer les conséquences. Cette taille pourrait être plus grande mais on "perd" en stationnarité au delà de 8.

Cela présente certes l'avantage de réduire la complexité de la transformation, mais aussi des inconvénients. Dans un premier lieu, la décorrélation est confinée à une surface restreinte, ce qui nuit à son efficacité lorsqu'un même objet ou une même texture chevauche plusieurs blocs. D'autre part, chacun des blocs étant codé de manière indépendante, ils peuvent subir des dégradations différentes au cours du processus de compression/décompression, différences qui vont sauter aux yeux en frontière des blocs reconstruits. C'est ce qu'on appelle les "effets de bloc".

### La transformée en ondelettes discrète : DWT

La décomposition en ondelettes a été abordée dans de nombreux articles (Mallat, 1999; Le Pennec et Mallat, 2005; Santoso *et al.*, 1997; Cui *et al.*, 2012), autant pour présenter les aspects mathématiques que les intérêts de ce type d'approche, notamment pour la compression.

Une transformée en ondelettes 2D possède trois fonctions d'ondelettes (Grgic *et al.*, 2001) : la première fonction d'ondelettes permet de récupérer les détails horizontaux, la deuxième les détails verticaux et enfin la troisième les détails diagonaux. La DWT 2D peut être implantée en utilisant un banc de filtres, illustré sur la FIGURE 2.2.

La propriété de *multirésolution* est parmi les propriétés les plus importantes des ondelettes. En effet, suite au passage de l'image à travers des filtres, on obtient la nouvelle image de la FIGURE 2.3. Cette image est décomposable en quatre parties distinctes par filtrage des coefficients de l'image, d'une part par le filtre  $a_L$  et d'autre part par le filtre  $a_H$ , suivi par la décimation d'un facteur 2. Cette dernière opération, aussi appelée *sous-échantillonnage d'un facteur 2*, est notée  $\downarrow 2$  et consiste de se débarrasser d'un coefficient sur deux.

L'algorithme peut être représenté par un banc de filtres d'analyse en quadrature miroir (Vaidyanathan, 1990), illustré dans la FIGURE 2.2. Ce banc de filtres d'analyse permet ainsi l'implantation effective de la transformée en ondelettes rapide. Chacune des sous

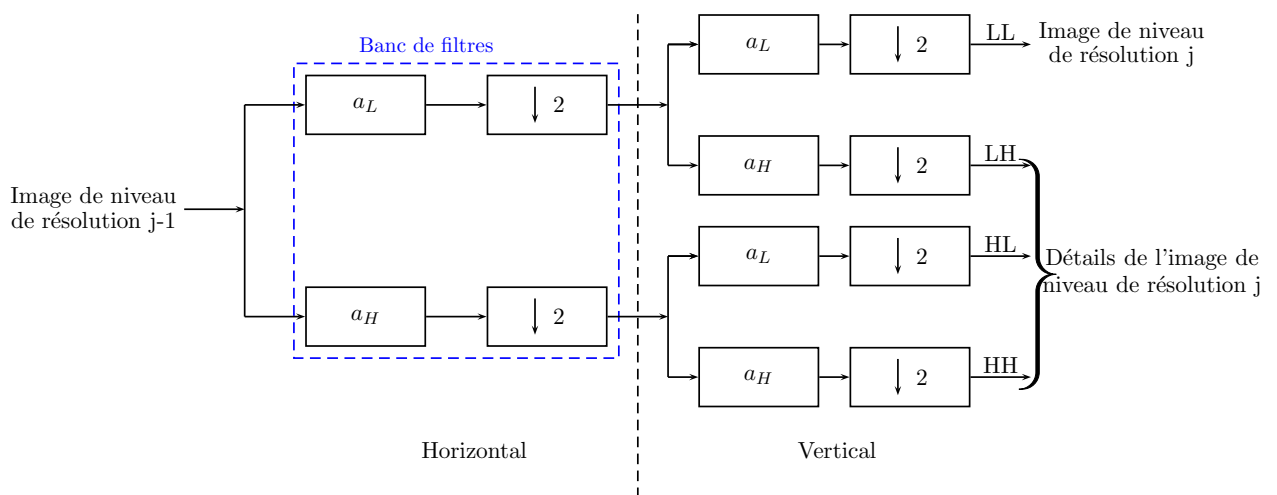


FIGURE 2.2: Transformée en ondelettes 2D. Schéma de décomposition classique en sous bandes, avec : LL, coefficients d'approximation ; HL, coefficients de détails verticaux ; LH, coefficients de détails horizontaux ; HH, coefficients de détails diagonaux

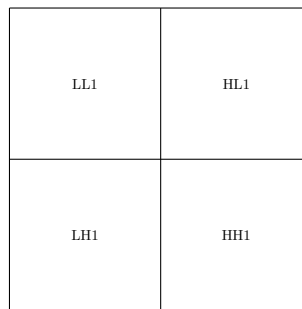


FIGURE 2.3: Image résultante après le premier niveau décomposition

images obtenues par filtrage contient une partie de l'information de l'image de départ. Les blocs LH, HL et HH contiennent tous des détails de l'image, ou informations hautes fréquences. L'image est analysée dans différentes directions : les détails horizontaux sont contenus dans LH, les détails verticaux dans HL et les détails diagonaux dans HH. Le bloc LL contient une image de plus basse résolution (4 fois moins de pixels que l'image initiale) dont les détails dans différentes directions ont été soustraits (principe de la multirésolution).

L'observation de l'image, en FIGURE 2.5, permet d'illustrer les transformations subies par l'image Lena FIGURE 2.4 lors de l'application de la DWT 2D.

Suite à ce premier niveau de décomposition, il est possible de poursuivre la décomposition en suivant l'algorithme de décomposition pyramidale de Mallat, comme montré dans FIGURE 2.6.

En vue d'obtenir la décomposition désirée, l'algorithme de pyramide de Mallat procède

FIGURE 2.4: Image Lena originale  $512 \times 512$ FIGURE 2.5: Décomposition DWT 1 niveau de l'image Lena  $512 \times 512$ 

à un balayage des lignes alterné avec un balayage des colonnes. En effet, le balayage des lignes conduit à décomposer l'image verticalement en des coefficients d'approximations à gauche et en des coefficients de détails à droite. Après chaque balayage des lignes, le balayage des colonnes permet de décomposer l'image horizontalement en deux parties approximation et détail.

On procède ainsi à une décomposition multirésolution de l'image qui s'illustre notamment par le fait qu'à chaque niveau de décomposition, le nombre de pixels de  $LL$  est divisé par 4. La résolution de l'image est donc divisée par 4. Le processus de décomposition et l'image résultante sont illustrés à la FIGURE 2.2.

La reconstruction d'une image à partir de ses coefficients en ondelettes prend une signification évidente : l'image, à sa résolution la plus grande, est égale à la somme d'une version floue, et des détails apparaissant à des échelles différentes, c'est à dire à des

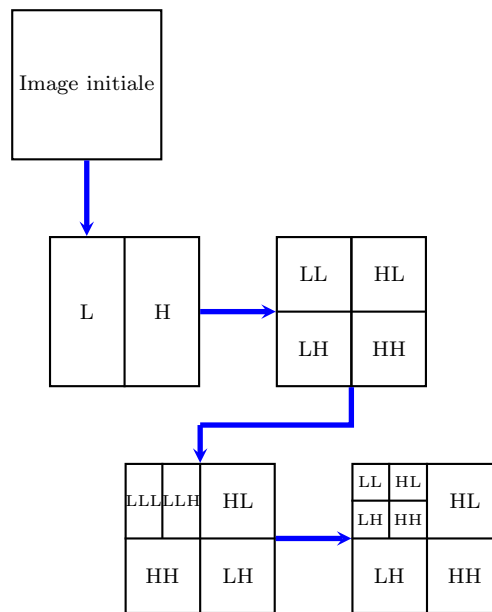


FIGURE 2.6: Décomposition pyramidale d'une image

résolutions différentes. De ce fait, la compression par ondelettes a été un réel progrès dans le domaine des images.

La transformée en ondelettes DWT présente l'avantage de faire ressortir à la fois les grandes variations et les détails de l'image analysée, ce que ne permet pas la DCT. Par ailleurs, elle permet de caractériser à la fois en temps et en fréquence un signal. En effet, dans une image, la DCT permet de caractériser globalement l'image en fréquence alors que la transformée en ondelettes va détecter les zones riches en informations (zones à fort contraste) et les sépare.

### La DWT *vs* La DCT

La DCT et la DWT présentent leurs avantages et leurs inconvénients. Mais le degré de complexité (au niveau de l'implantation), la consommation de l'énergie (le coût de transmission via le RCSF) restent, dans notre cas, les critères les plus importants qu'il faut prendre en considération.

Plusieurs travaux ont utilisé des algorithmes de compression, basés sur la DCT pour le transfert des images à travers les réseaux de capteurs sans fil pour plusieurs raisons. Parmi ces raisons, on cite principalement, la simplicité de la DCT par rapport à la DWT.

Les contributions (Veres *et al.*, 2001; Wu et Abouzeid, 2004a), montrent que la transformée en ondelettes offre une meilleure qualité d'image à un taux de compression donné par rapport à la DCT. Cependant, les performances des systèmes de compression à base

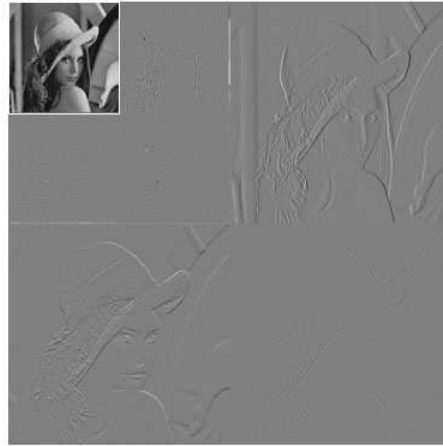


FIGURE 2.7: Décomposition DWT 2 niveaux de l'image Lena  $512 \times 512$

de transformée en ondelettes semblent moins adaptés pour les applications des réseaux de capteurs sans fil puisqu'ils nécessitent davantage des ressources mémoire et consomment plus d'énergie.

Comme les noeuds capteurs sont par nature très limités en capacité de calcul et de mémoire, on s'intéressera dans ce travail uniquement à la transformée en cosinus discrète. Notons que même si celle-ci présente l'inconvénient des effets de blocs, différentes approches ont été proposées (Jung *et al.*, 1998; Minami et Zakhor, 1995) pour réduire cet inconvénient.

### 2.2.2 Quantification

La matrice transformée occupe autant ou plus d'espace de stockage que la matrice de pixel en entrée. Pour cela on est obligé d'introduire une nouvelle opération qui aura pour but de réduire les valeurs de la matrice transformée. Cette opération est appelée *quantification*.

La quantification consiste en l'observation d'un nombre  $x$  et en la sélection de son approximation dans un ensemble fini de valeurs prédéterminées. Le quantificateur est donc une fonction  $Q = \mathbb{R} \rightarrow C$ , où  $\mathbb{R}$  est l'ensemble des nombres réels et  $C = \{w_1, w_2, \dots, w_N\} \subset \mathbb{R}$  est l'ensemble des sorties, nommé dictionnaire (*codebook* en anglais),  $w_1, w_2, \dots, w_N$  étant les mots de code.

Dans le schéma de compression d'image, la quantification est le processus de réduction du nombre de bits nécessaire au stockage d'une valeur entière par diminution de la précision de l'entier. Cette étape dégrade de manière irréversible l'image.

La quantification peut s'effectuer échantillon par échantillon, dans ce cas on parlera

de quantification scalaire (QS), ou bloc par bloc, c'est ce qu'on appelle la quantification vectorielle (QV). La QV consiste donc à traiter conjointement un ensemble de  $n$  échantillons.

Pour l'élaboration d'un algorithme de quantification, on fait normalement appel aux études permettant de mieux estimer les limites de la vision. On sait que, lors de la reconstruction d'une image, les hautes fréquences sont moins importantes que les basses fréquences. À ce niveau, ces éléments jugés peu significatifs visuellement sont plus grossièrement quantifiés en utilisant un pas de quantification plus grand.

### La quantification scalaire (QS)

Il est possible de réduire encore plus la précision des coefficients par quantification scalaire (Gersho et Gray, 1992) au fur et à mesure qu'on s'éloigne du point d'origine (à la position  $(0, 0)$  de la matrice à quantifier); car plus on s'éloigne, moins la contribution de ces coefficients à la représentation de l'image est importante. Donc la quantification revient à utiliser une matrice de quantification où pour chaque position  $(i, j)$  de coefficient dans la matrice DCT, correspond une valeur dans la matrice de quantification.

Ces valeurs seront de plus en plus grandes lorsqu'on s'éloigne de la position  $(0, 0)$ , c'est-à-dire de la composante continue de la matrice DCT. Après la transformée, tous les coefficients sont quantifiés comme montré dans l'équation 2.5.

$$Valeur_{\text{quantifiée}}(i, j) = Round \left( \frac{DCT(i, j)}{quantification(i, j)} \right) \quad (2.5)$$

Ce processus de quantification convertit les coefficients représentés en nombre réel en nombre entier par la fonction *Round* qui arrondit à l'entier le plus proche.

Il faut noter que l'utilisation des valeurs de quantification très élevées génère des grandes erreurs pendant le processus de déquantification.

L'intérêt de la quantification scalaire est sa flexibilité par variation du pas de quantification. Ce type de quantificateur est très simple à mettre en oeuvre du fait du réglage d'un seul paramètre pour obtenir le débit souhaité. Sa complexité calculatoire est de 2 opération (1 division + 1 arrondi) pour un coefficient et donc de :  $O(n) = 2n$  opérations pour un bloc de  $n$  coefficients.

### La quantification vectorielle (QV)

La quantification vectorielle est une généralisation de la quantification scalaire (QS) à des vecteurs. Elle a été introduite par Gersho et Gray (Gersho, 1982; Gersho et Gray,

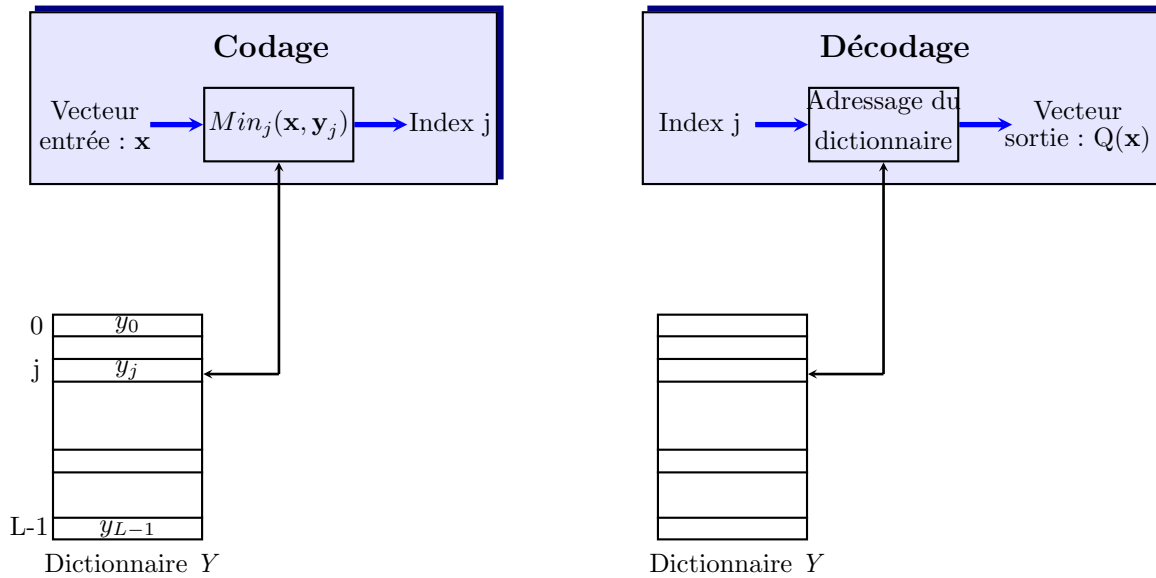


FIGURE 2.8: Principe de quantificateur vectoriel

1992) et s'est avérée performante pour la compression des images. Son principe est de coder un ensemble de  $n$  échantillons (vecteur) au lieu de coder chacun de ces échantillons individuellement.

Comme illustré dans la FIGURE 2.8, le codage s'effectue en approximant chaque vecteur  $x_i$  du signal d'entrée par l'indice du vecteur de code le plus ressemblant dans le dictionnaire  $Y$  avec une distorsion minimale entre  $x$  et  $y_j$  de  $Min_j(x, y_j)$ .

Le dictionnaire utilisé, dont le contenu est calculé à partir d'une séquence d'apprentissage par l'algorithme *LBG* (Linde *et al.*, 1980; Gersho et Gray, 1992), détermine les "meilleurs représentants" pour un ensemble de signaux donné.

L'algorithme *LBG* crée un code et effectue sa classification sur une séquence d'apprentissage constituée par des vecteurs appartenant à différentes images. Ces vecteurs sont indexés et lors du codage, c'est uniquement l'adresse du code vecteur  $y_i$  ainsi sélectionné qui sera transmise ou stockée. Cette méthode est connue sous le nom de *quantification vectorielle non structurée*.

Pour reconstituer le vecteur, le décodeur dispose d'une réplique du dictionnaire  $Y$  et consulte celui-ci pour fournir le code vecteur d'indice correspondant à l'adresse reçue.

La complexité calculatoire, une fois que le dictionnaire est construit, du codage d'un échantillon de  $n$  éléments est de :  $3nL - L$  opérations où  $L$  est la taille du dictionnaire.

Un cas particulier de la quantification vectorielle non structurée est la quantification vectorielle structurée ou algébrique (Moureaux, 2007). Cette méthode présente l'avantage d'éviter la génération et le stockage du dictionnaire. L'étape de quantification a donc une complexité nettement plus faible que la QV par apprentissage : elle répond à des opérations



mathématiques simples et rapides de type "arrondi". En contrepartie, ses performances sont plus faibles que celles de la QV non structurée (LBG).

Les auteurs dans (Moureaux *et al.*, 1998), ont présenté une méthode de quantification vectorielle avec une faible complexité calculatoire qui est de :  $O(n) = 2n^2 + 6n - 10$  opérations/vecteur sur le réseau  $Z^n$  et qui ne dépend pas de la taille dictionnaire, comme c'est le cas pour les méthodes par apprentissage, mais simplement de la dimension  $n$  des vecteurs.

Un des résultats fondamentaux des travaux de Shannon concernant la relation débit/distorsion montre que l'on obtient de meilleures performances en utilisant la quantification vectorielle. En revanche, la plupart des approches vectorielles demeurent coûteuses en calcul même si la quantification vectorielle algébrique permet de réduire considérablement ce coût. Elles nécessitent, par ailleurs, beaucoup de ressources mémoire. Pour toutes ces raisons, elles ne sont pas actuellement éligibles dans le contexte des réseaux de capteurs.

### 2.2.3 Codage entropique

Le codage entropique est employé à la dernière étape de la chaîne de compression avec pertes (voir FIGURE 2.1) afin d'exploiter les redondances présentes à la sortie du quantificateur.

Le codage entropique consiste à coder en binaire chaque valeur des coefficients transformés et quantifiés avec un code à longueur variable afin de diminuer le débit de transmission. En effet, un tel codage exploite les propriétés statistiques des coefficients quantifiés en utilisant des codes courts pour représenter les événements les plus probables et des codes plus longs pour les occurrences rares.

Les codes entropiques sont basés sur la génération de mots dont la longueur dépend de la probabilité d'apparition des symboles de la source qu'il représente : un grand nombre de bits sera utilisé pour coder un symbole peu probable tandis qu'un symbole redondant sera codé sur très peu de bits. 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, 0101 ne l'est pas : 0 est le préfixe du mot 0101 et 10 est le préfixe de 101.

Il existe de nombreux codes entropiques, parmi lesquels le code de Huffman (Huffman, 1952), le codage arithmétique et le codage de Golomb, qui sont décrit brièvement ci-dessous.

Afin de mesurer la quantité d'informations codées dans une image, on utilise le terme

entropie. Plus l'entropie d'une image est grande, plus elle contient d'informations (Nelson et Soulard, 1993).

Si la probabilité des composantes de l'image était connue, il devrait exister une manière de coder les symboles, de façon à ce que l'image prenne moins de place. Parmi ces méthodes, on cite : Huffman et le codage arithmétique (Gutter, 1995) , (Nelson et Soulard, 1993).

### Codage de Huffman

L'algorithme de codage de Huffman (Huffman, 1952) est un algorithme de codage par bloc, c'est-à-dire que les symboles (groupe de symboles) sont représentés à l'aide d'un code binaire à longueur variable (Huffman, 1952). Plus une valeur est probable, plus court est le mot associé. Tout le problème consiste évidemment à trouver le jeu de codes optimal afin de minimiser le coût de codage moyen par symbole.

L'algorithme considère chaque message à coder comme étant une feuille d'un arbre qui reste à construire. L'idée est d'attribuer aux deux messages ayant les plus faibles probabilités, les mots codés les plus longs. Ces deux mots codés ne se différencient que par leur dernier bit.

La construction de l'arbre de Huffman se fait en ordonnant dans un premier temps les valeurs des coefficients quantifiés par fréquence d'apparition. Successivement les deux coefficients de plus faible fréquence d'apparition sont retirés de la liste et rattachés à un nœud dont le poids vaut la somme des fréquences des deux symboles. Le coefficient de plus faible poids est affecté à la branche 1, l'autre à la branche 0 et ainsi de suite en considérant chaque nœud formé comme un nouveau coefficient, jusqu'à obtenir un seul nœud parent appelé racine.

Le code de chaque coefficient correspond à la suite des codes le long du chemin allant de la racine à ce coefficient. Ainsi, plus le coefficient est "profond" dans l'arbre, plus le mot de code sera long.

Cet algorithme admet cependant un inconvénient : il repose sur la connaissance complète de la source à coder (il existe cependant des versions adaptatives). D'autre part, il ne permet pas d'atteindre des débits inférieurs à 1 bit/échantillon.

De plus, pour décompresser, il faut connaître les codes et donc la table, qui est ajoutée à l'image compressée, ce qui diminue le taux de compression, surtout pour les petites images. Ce problème est éliminé par le codage Huffman adaptatif, qui modifie sa table au fil du traitement et peut donc démarrer avec une table de base. Il est alors intéressant de se baser sur la table par défaut utilisé dans le format JPEG. Cela permet d'atteindre

un meilleur taux de compression car dans ce cas il n'est pas nécessaire d'insérer la table créée dans l'image comprimée.

### Codage arithmétique

Contrairement aux algorithmes de codage utilisant des mots de codes à longueur variable qui associe à des symboles des motifs binaires dont la taille dépend de leur distribution, le codeur arithmétique (Witten *et al.*, 1987) traite l'image dans son ensemble, en lui associant un unique nombre décimal rationnel.

Ce nombre décimal, compris entre 0 et 1, dépend non seulement des coefficients quantifiés dans l'ordre où ils apparaissent, mais aussi de leur distribution statistique.

Alors que les codes d'Huffman sont à longueur entière, le codage arithmétique, optimal au niveau du bit, permet de coder des symboles sur un nombre de bits arbitraire. Pour comparer le codage arithmétique au codage de Huffman, on suppose que l'on va coder des symboles de probabilités 99% et 1%. L'information contenant le symbole dont la probabilité est  $P$  est codée par  $-\log(P)$  bits, c'est-à-dire que le symbole qui a une probabilité de 99% sera codé sur 0,015 bits, tandis que le codage de Huffman utilise au minimum 1 bit par symbole (Witten *et al.*, 1999).

Le codage arithmétique est très dépendant des symboles précédemment encodés, ce qui le rend peu robuste aux erreurs de transmission, et empêche le décodage à partir d'une position aléatoire dans le flux de données.

Il existe de nombreux codeurs arithmétiques, tous basés sur le même principe, parmi lesquels on peut citer le QM-Coder utilisé par JPEG2000 (Min *et al.*, 2007) qui utilise son propre modèle adaptatif et qui réduit le temps de calcul en supprimant les multiplications par des approximations, au détriment d'une légère perte de compression. On peut également citer CABAC (Context-Adaptive Binary Arithmetic Coding) (Marpe *et al.*, 2003) utilisé par H.264/MPEG-4 AVC.

### Codage de Golomb

Dans le principe, un codage Golomb est assez similaire à un code Huffman. La principale différence vient du fait qu'on suppose que les variables d'entrée suivent une certaine loi de probabilité qui va implicitement induire le codage. Cette propriété permet de ne pas avoir besoin de calculer la table de correspondances des symboles, et le rend ainsi facilement adaptatif (Weinberger *et al.*, 1996; Taubman *et al.*, 2002; Malvar, 2006).

Un codeur Golomb (Golomb, 1966) de paramètre  $m$  noté  $G_m$  encode une valeur entière positive  $n$  en deux parties : une représentation binaire de  $n \bmod m$  et une représentation

unaire de la partie entière par défaut de la division de  $n$  par  $m$  ( $\lfloor \frac{n}{m} \rfloor$ ). Pour que le codage soit non ambigu, la représentation unaire est une suite de bits à 1 suivie d'un bit d'arrêt à 0. Les codeurs Golomb-Rice (Rice et Plaunt, 1971; Gallager et Van Voorhis, 1975) sont un cas particulier du codeur Golomb avec  $m = 2k$ , particulièrement adaptés pour le codage/décodage informatique. Ainsi les divisions se font par simple décalage de bits et les modulo par une opération logique utilisant un masque.

### Quelle codeur entropique utiliser ?

L'avantage du codage de Huffman est la rapidité : avec une table indexée par les symboles d'entrée, le temps de codage dépend uniquement de la recopie de la séquence associée au symbole d'entrée. Cependant, le codage de Huffman nécessite un recalcul complet du code de sortie à chaque phase d'adaptation, ce qui est coûteux en temps de calcul.

L'avantage de codeur de Golomb est la rapidité aussi : il nécessite uniquement un modulo pour obtenir les symboles représentant la partie de poids faible des symboles d'entrée, et une division suivie du codage unaire. Comme pour les codes de Huffman, il y a une perte entropique due à l'encodage. Cette perte peut cependant être diminuée à l'aide de systèmes adaptatifs (Weinberger *et al.*, 1996).

Le codeur de Golomb nécessite beaucoup moins de calculs que le codeur de Huffman pour être rendu adaptatif. En effet l'adaptation du code Huffman nécessite le recalcule de l'arbre complet, tandis que pour le code Golomb seul le paramètre  $m$  doit être mis à jour.

Le codeur arithmétique permet d'atteindre un bon taux de compression, de plus il est facilement adaptatif, sans avoir à modifier l'algorithme de codage en lui même, mais le modèle qu'il utilise. L'inconvénient du codeur arithmétique est qu'il demande un peu plus de temps de calcul que les codeurs de Huffman et Golomb. Le codeur arithmétique doit effectuer quelques opérations (au minimum 2 multiplications, 1 division, 3 additions et 2 soustractions pour chaque symbole dans l'implantation de (Bodden *et al.*, 2007)) afin de mettre à jour les bornes de l'intervalle.

Comme montré dans les paragraphes précédents, chaque codeur a des avantages et des inconvénients. Alors **quel codeur entropique faut-il utiliser ?**

Le cadre d'application guide généralement le choix du codeur entropique. Si la compression doit favoriser la rapidité et qu'il est possible de modéliser une ou plusieurs distributions de probabilités représentatives des coefficients de la matrice transformée, les codes de Huffman sont souvent préférés.

Les codes Golomb-Rice adaptatifs sont réputés pour fournir un bon compromis entre

rapidité et taux de compression lorsqu'il s'agit d'images naturelles, après l'étape de décorrélation.

Si l'application est moins focalisée sur la vitesse d'exécution que sur les performances de compression, les codeurs arithmétiques adaptatifs sont privilégiés.

Si l'application est focalisée sur la vitesse d'exécution ainsi que sur les performances de compression (c'est l'orientation qui a été prise pour cette thèse), le codage de Golomb couplé avec le codeur arithmétique  $QM$  présente un bon compromis complexité calculatoire/taux de compression. L'utilisation du codage de Golomb et le codeur  $QM$  au lieu du codage de Huffman est intéressante en termes de complexité algorithmique et espace mémoire (Howard et Vitter, 1992).

### 2.2.4 Synthèse

Dans les paragraphes précédents, nous avons présenté les trois étapes de la chaîne de compression. A chaque étape, notre choix s'oriente vers la méthode la plus adaptée.

Même si la complexité de la DCT est inférieure à celle de la transformées en ondelettes, la DCT demeure coûteuse pour une implantation dans le contexte des réseaux de capteurs.

Dans le schéma de compression basé sur la DCT, l'étape de transformée représente à elle seule 60% du coût total en énergie de la compression (Marcelloni et Vecchio, 2008).

Cependant, afin de rendre l'utilisation de la DCT compatible avec les contraintes des réseaux de capteurs, nous avons étudié **les différents algorithmes de DCT rapides** existant dans littérature et proposé **une approche hybride adaptée au cas des réseaux de capteurs**.

## 2.3 Réduction DCT

Plusieurs méthodes pour réduire le calcul de la transformée en cosinus discrète sont présentes dans la littérature. Les deux approches principales qui servent à calculer la DCT avec une complexité réduite par rapport à l'équation de référence 2.4 sont **la DCT réduite** et **la DCT rapide** (voir FIGURE 2.9).

L'approche de la DCT réduite permet de réduire le nombre de coefficients DCT à calculer alors que la DCT rapide réduit le nombre d'opérations pour calculer les coefficients de la DCT.

Le tableau 2.1 présente les algorithmes de référence en calcul de la DCT rapide. La comparaison de l'ensemble de ces algorithmes est décrite dans la section 2.3.3.

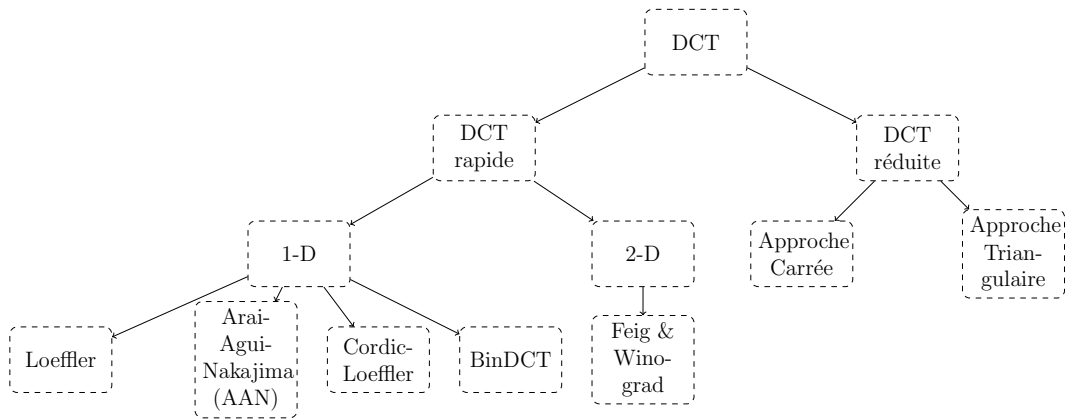


FIGURE 2.9: Les méthodes de calcul de la DCT

	Coût de la DCT 1-D			Coût de la DCT 2-D		
	Mult.	Add.	Décal.	Mult.	Add.	Décal.
Loeffler (Loeffler <i>et al.</i> , n.d.)	11	29	-	176	464	-
AAN (ARAI <i>et al.</i> , 1988)	5	29	-	80	464	-
Feig & Winograd (Feig et Winograd, 1992)	-	-	-	54	462	-
BinDCT (Tran, 2000)	-	36	19	-	576	304
Cordic (Sun <i>et al.</i> , 2007)	-	38	16	-	608	256

TABLE 2.1: Coût de calcul d'un bloc  $8 \times 8$  des différents algorithmes pour le calcul de la DCT

Dans la première et la deuxième section de ce chapitre (sec. 2.3.1) une brève description de l'approche de factorisation de la matrice et l'approche de DCT réduites est donnée, puis les algorithmes de calcul de la DCT rapide combinés avec l'approche de la DCT réduite sont détaillés. Enfin, le processus de compression utilisé pour permettre une économie d'énergie du nœud caméra est proposé.

L'objectif de cette section est de répondre à la question :

*Quel algorithme pour le calcul de la DCT utiliser afin d'augmenter la durée de vie du nœud caméra tout en préservant une bonne qualité d'image à la réception c'est-à-dire qui optimise le compromis distorsion-compléxité ?*

### 2.3.1 L'approche par factorisation de la matrice (DCT rapide)

Cette approche consiste à réécrire la transformée sous forme matricielle et à factoriser la décomposition de manière à réduire le nombre de multiplications scalaires nécessaires.

En d'autres termes, on peut réduire le coût de calcul de la DCT initialement de 4096 additions et 8192 multiplications (équation référence 2.4) à 1024 multiplications et 1024 additions, pour un bloc  $8 \times 8$  en ne calculant qu'une seule fois le produit des deux cosinus. En effet, il n'y a que 64 combinaisons possibles pour le couple  $(u,v)$  donc il suffit de calculer au départ les 64 valeurs possibles, les stocker et ensuite, on ne les calcule plus.

Donc cela revient à créer, comme montré dans l'équation (2.6), une matrice  $A$  :

$$A(u, v) = \begin{cases} \sqrt{\frac{2}{N}} \cos\left(\frac{\pi(2v+1)u}{2N}\right) & \text{si } 1 < u < N \\ \frac{1}{\sqrt{N}} & \text{si } u = 0 \end{cases} \quad (2.6)$$

Une fois cette matrice créée, on crée la matrice transposée. Ensuite la transformée par bloc  $8 \times 8$  se réduit à deux multiplications matricielles :

$$DCT = A_{N \times N} \cdot \text{Img}_{N \times N} \cdot A_{N \times N}^T$$

De plus, en se basant sur la symétrie de :  $a_x^u = \frac{2}{N} C(u) \cos\left(\frac{\pi(2x+1)u}{2N}\right)$  et en supposant que  $N = 8$ , on peut réécrire la fonction de DCT 1-D comme suit :

$$F(u) = \begin{cases} \sum_0^3 a_x^u (f(x) + f(7-x)) & \text{si } u \text{ est pair} \\ \sum_0^3 a_x^u (f(x) - f(7-x)) & \text{si } u \text{ est impair} \end{cases}$$

On peut présenter les expressions sous la forme matricielle comme suit :

$$\begin{pmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{pmatrix} = C_j \cdot \begin{pmatrix} C_0 & C_0 & C_0 & C_0 & C_0 & C_0 & C_0 & C_0 \\ C_1 & C_3 & C_5 & C_7 & -C_7 & -C_5 & C_3 & -C_1 \\ C_2 & C_6 & -C_6 & -C_2 & -C_2 & -C_6 & C_6 & C_2 \\ C_3 & -C_7 & -C_1 & -C_5 & C_5 & C_1 & C_7 & -C_3 \\ C_4 & -C_4 & -C_4 & C_4 & C_4 & -C_4 & -C_4 & C_4 \\ C_5 & -C_1 & C_7 & C_3 & -C_3 & -C_7 & C_1 & -C_5 \\ C_6 & C_2 & C_2 & C_6 & C_6 & C_2 & C_2 & C_6 \\ C_7 & -C_5 & C_3 & -C_1 & C_1 & -C_3 & C_5 & -C_7 \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{pmatrix} \quad (2.7)$$

Où  $C_k = \cos\left(\frac{k\pi}{16}\right)$ .

Le calcul d'un seul vecteur de coefficients nécessite 64 multiplications et 56 additions. Dans le cas du calcul du bloc  $8 \times 8$ , le nombre d'opérations est de 1024 multiplications et

896 additions. Cependant, on peut remarquer que les coefficients sont symétriques pour les lignes d'index pairs et antisymétriques pour celle d'index impairs. Ce qui présente un grand avantage.

La principale idée, ici, est de tirer avantage de cette symétrie et réduire le nombre de multiplications de 64 à 32.

En réécrivant la transformée 2.4 sous forme matricielle tout en factorisant la décomposition, on obtient la matrice suivante :

$$\begin{pmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{pmatrix} = C_j \cdot \begin{pmatrix} C_0 & C_0 & C_0 & C_0 & 0 & 0 & 0 & 0 \\ C_2 & C_6 & -C_6 & -C_2 & 0 & 0 & 0 & 0 \\ C_4 & C_4 & -C_4 & C_4 & 0 & 0 & 0 & 0 \\ C_6 & -C_2 & -C_2 & -C_6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & C_1 & C_3 & C_5 & C_7 \\ 0 & 0 & 0 & 0 & C_3 & -C_7 & -C_1 & -C_5 \\ 0 & 0 & 0 & 0 & C_5 & -C_1 & C_7 & C_3 \\ 0 & 0 & 0 & 0 & C_7 & -C_5 & C_3 & -C_1 \end{pmatrix} \begin{pmatrix} x_0 + x_7 \\ x_1 + x_6 \\ x_2 + x_5 \\ x_3 + x_4 \\ x_4 - x_7 \\ x_5 - x_6 \\ x_6 - x_5 \\ x_7 - x_4 \end{pmatrix} \quad (2.8)$$

La matrice ci-dessus peut encore être simplifiée par les 2 sous matrices de 2.9 et 2.10.

$$\begin{pmatrix} y_0 \\ y_2 \\ y_4 \\ y_6 \end{pmatrix} \begin{pmatrix} C_4 & C_4 & C_4 & C_4 \\ C_2 & C_6 & -C_6 & -C_2 \\ C_4 & C_4 & -C_4 & C_4 \\ C_6 & -C_2 & -C_2 & -C_6 \end{pmatrix} \begin{pmatrix} x_0 + x_7 \\ x_1 + x_6 \\ x_2 + x_5 \\ x_3 + x_4 \end{pmatrix} \quad (2.9)$$

$$\begin{pmatrix} y_1 \\ y_3 \\ y_5 \\ y_7 \end{pmatrix} \begin{pmatrix} C_1 & C_3 & C_5 & C_7 \\ C_3 & -C_7 & -C_1 & -C_5 \\ C_5 & -C_1 & C_7 & C_3 \\ C_7 & -C_5 & C_3 & -C_1 \end{pmatrix} \begin{pmatrix} x_4 - x_7 \\ x_5 - x_6 \\ x_6 - x_5 \\ x_7 - x_4 \end{pmatrix} \quad (2.10)$$

Cette nouvelle décomposition de la matrice DCT sert à diminuer de façon importante (de 75%) le calcul de 1024 multiplications à 256 multiplications uniquement (32 multiplications pour chaque ligne de 8 valeurs) dans le traitement d'un bloc image  $8 \times 8$ .

En se basant sur le principe de la factorisation de la matrice, la matrice DCT  $T$  peut être décomposée en produits de sous matrices. En d'autres termes,  $T$  peut s'écrire sous la forme  $T = T_1 \times T_2 \times K \times T_k$ .





$$D = \begin{pmatrix} C_4 & S_4 & 0 & 0 & 0 & 0 & 0 & 0 \\ S_4 & -C_4 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & C_6 & S_6 & 0 & 0 & 0 & 0 \\ 0 & 0 & -S_6 & C_6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & C_1 & 0 & 0 & S_1 \\ 0 & 0 & 0 & 0 & 0 & C_3 & S_3 & 0 \\ 0 & 0 & 0 & 0 & 0 & -S_3 & C_3 & 0 \\ 0 & 0 & 0 & 0 & -S_1 & 0 & 0 & C_1 \end{pmatrix} \quad (2.14)$$

$$E = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \quad (2.15)$$

Ainsi, le graphe de flux de l'algorithme de Chen obtenu à partir des factorisations de la matrice transformée, est présenté dans la figure 2.10.

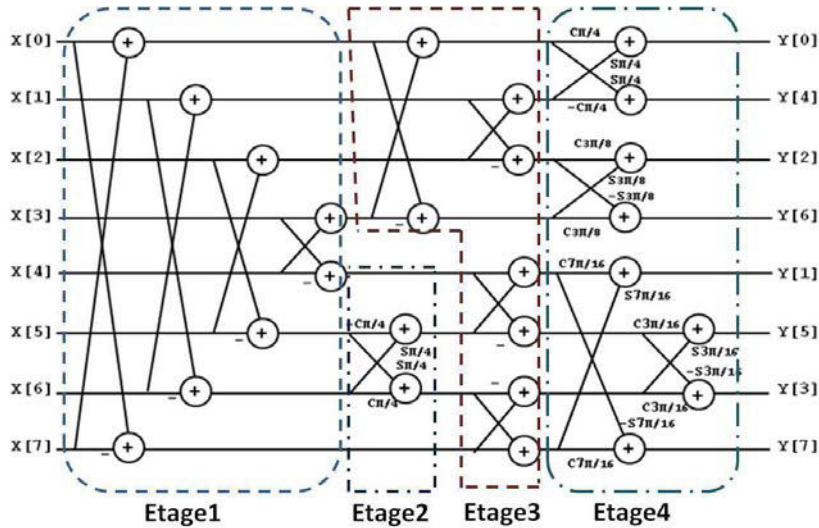


FIGURE 2.10: Graphe des flux de Chen

Chaque étage du graphe de Chen présente une matrice : l'étage 1 correspond à la matrice  $A$ , l'étage 2 à la matrice  $B$ , l'étage 3 à la matrice  $C$  et l'étage 4 à la matrice  $D$ .

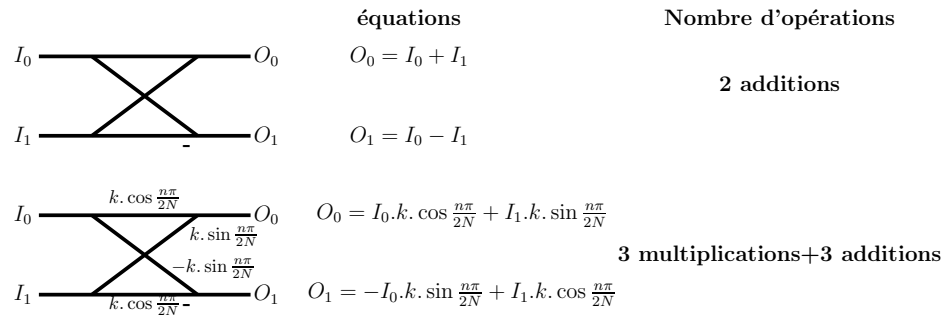


FIGURE 2.11: Les symboles utilisés pour les graphes de flux

La DCT 2-D peut être obtenue en appliquant d'abord la DCT 1-D sur les lignes du bloc de pixels puis sur les colonnes du bloc résultant.

On constate que l'algorithme de Chen utilise 16 multiplications et 26 additions pour le calcul de la DCT 1-D. Le nombre d'opérations de l'algorithme de Chen peut être diminué de 3 multiplications mais avec une augmentation de nombre d'additions à 29 additions par l'utilisation de la méthode de calcul de rotation présentée par l'équation 2.26.

Les rotations peuvent être calculées avec seulement 3 multiplications et 3 additions au lieu de 4 multiplications et 2 additions par l'utilisation de l'équivalence montrée dans l'équation 2.16.

$$\begin{cases} y_0 = a.x_0 + b.x_1 = (b - a).x_1 + a.(x_0 + x_1) \\ y_1 = -b.x_0 + a.x_1 = -(b + a).x_0 + a.(x_0 + x_1) \end{cases} \quad (2.16)$$

Cela permet de diminuer le nombre de multiplications mais augmente le nombre d'additions de l'algorithme de Chen. Cela permet de diminuer la complexité de l'algorithme puisque les opérations ne prennent pas le même temps (les additions sont moins coûteuses que les multiplications).

Le nombre d'opérations de l'algorithme de Chen, peut être optimisé donc à 13 multiplications et 29 additions (Soit 208 multiplications et 416 additions en 2-D). Cela fait une économie d'environ 80% en nombre de multiplications et 50% en nombre d'additions par rapport à la DCT référence 2.4.

### 2.3.2 L'approche zonale (DCT réduite)

Les algorithmes de calcul de la DCT basés sur l'approche de factorisation de matrice, génèrent, à partir de  $N$  entrées, le même nombre de sorties ( $N$  coefficients DCT). Cependant, dans la plupart des applications de traitement d'image, l'information importante de l'image est essentiellement portée par les coefficients de basse fréquence. Ces derniers sont peu nombreux et se trouvent essentiellement dans la partie supérieure gauche du bloc transformé par la DCT. Ainsi, c'est seulement ces coefficients qui pourraient être calculés. Cette méthode de calcul partiel est généralement mentionnée dans la littérature par "DCT réduite".

L'approche zonale consiste donc à opérer sur ce sous-ensemble des coefficients, c'est-à-dire ceux les plus importants, pour calculer, quantifier, encoder et transmettre seulement ceux là (Christopoulos *et al.*, 1995; Skodras, 1994; Wang, 1991).

Les auteurs dans (Jung *et al.*, 1996) ont proposé une approximation du bloc de DCT  $8 \times 8$  à partir des quatre premiers coefficients de basse fréquence. Le même principe a été repris par (Girod et Stuhlmuller, 1998) mais l'approximation du bloc  $8 \times 8$  de la DCT a été faite à partir de trois coefficients (la composante continue (DC) et de deux coefficients de hautes fréquences (AC)).

Les auteurs dans (Bo et Xuelong, 1999) ont proposé un schéma de classification des blocs de la DCT dans lequel pour chaque bloc d'entrée on vérifie s'il produit un bloc de zéro après quantification. Cette classification se fait en comparant la valeur absolue de la sommation des valeurs du bloc à un seuil.

(Pao et Sun, 1999; Lengwehasatit et Ortega, 2004) ont présenté un algorithme de classification qui se base sur le même principe que celui proposé par (Bo et Xuelong, 1999). (Pao et Sun, 1999; Lengwehasatit et Ortega, 2004) propose, après l'estimation des coefficients qui ont une grande probabilité d'être différents de zéro, de classer les blocs DCT selon cinq classes :

- tout le bloc DCT est à zéro
- le bloc est approximé à partir de la composante continue seulement
- le bloc est approximé à partir des deux premier coefficients de basse fréquence
- le bloc est approximé à partir d'un bloc de coefficients de taille  $4 \times 4$ .
- le bloc est approximé à partir de tous les coefficients de la DCT

Ces méthodes n'apportent pas un grand changement dans la complexité de calcul dans le cas où la majorité des blocs exige le calcul de tous les coefficients du bloc  $8 \times 8$  DCT.

(Xiuhua et al., 2011) ont proposé une autre approche qui consiste à prédire avant le

calcul de la DCT si le coefficient aboutira à zéro après quantification ou non. L'ordre de calcul des coefficients de la DCT est montré dans la table 2.2 :

1	1	1	1	3	5	7	9
1	1	1	2	4	6	8	10
1	1	2	4	6	8	10	11
1	2	4	6	8	10	11	12
3	4	6	8	10	11	12	13
5	6	8	10	11	12	13	14
7	8	10	11	12	13	14	15
9	10	11	12	13	14	15	16

TABLE 2.2: Les poids des coefficients de la DCT

Les coefficients, de la table 2.2, dont le poids correspondant est 1 sont calculés directement sans aucune prédiction et les coefficients dont le poids est supérieur à 1, ont besoin d'une prédiction afin de déterminer s'ils seront calculés.

Dans (Bracamonte *et al.*, 1996), les auteurs proposent une autre approche zonale qui consiste à sélectionner une zone de coefficients qui correspond au carré de longueur  $k$ , avec  $k < 8$ , situé dans la partie supérieure gauche du bloc  $8 \times 8$ . On a donc  $k^2$  coefficients à calculer et les autres sont négligés (ils seront implicitement mis à zéro).

Dans (Mammeri *et al.*, 2008), le même principe est repris, mais cette fois-ci, la zone de sélection des coefficients est définie par un triangle rectangle de cathète  $k$ , situé dans la partie supérieure gauche de la zone carrée précédemment mentionnée.

Comme montré dans la FIGURE 2.12, dans la forme triangulaire, il y a seulement  $\frac{k(k+1)}{2}$  coefficients significatifs au lieu de  $k^2$ . Et plutôt que de calculer  $k^2$  coefficients, il est possible de diminuer le nombre de coefficients de sortie à calculer pour chaque ligne. Ainsi, pour calculer les coefficients de la zone triangulaire nous avons proposé dans la section 2.3.3 d'appliquer les algorithmes de calcul de la DCT rapide avec  $N$  entrées et  $k$  sorties pour chaque ligne, puis pour chaque colonne  $i$  (on considère que  $i$  est initialisé à 1). nous appliquons l'algorithme avec  $N$  entrées et  $k - (i - 1)$  sorties.

Pour une valeur de  $k$  donnée, la forme triangulaire va permettre de diminuer le temps de calcul pour les étapes de quantification et de codage des coefficients par rapport à la forme carrée, puisqu'il y a moins de coefficients à traiter.

Comme les deux approches présentées, approche par factorisation de la matrice et approche par sélection zonale des coefficients sont complémentaires, nous avons proposé de les combiner. Nous désignerons une telle combinaison comme étant une DCT zonale rapide.

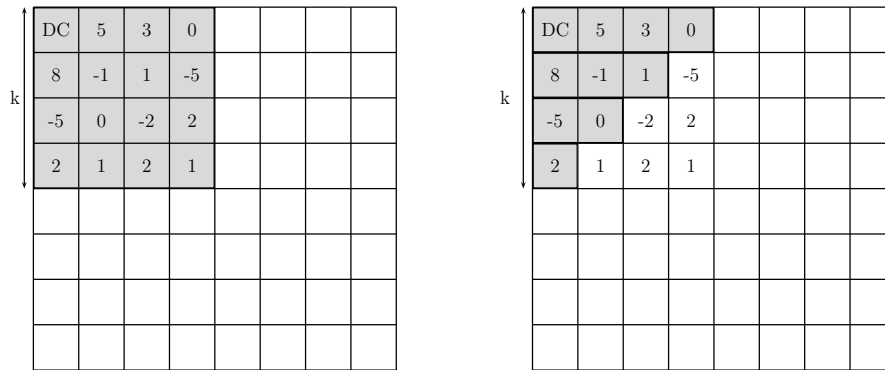


FIGURE 2.12: Différentes formes de DCT zonale ( $k=4$ ) a) Forme carrée b) Forme triangulaire

### 2.3.3 La DCT zonale rapide

Le principe de la DCT zonale rapide consiste à sélectionner la zone des coefficients de la DCT à coder dans le bloc et les calculer en se basant sur des algorithmes de DCT 1-D rapides comme les algorithmes LLM, Cordic Loeffler, BinDCT ou AAN. Cela réduit 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.

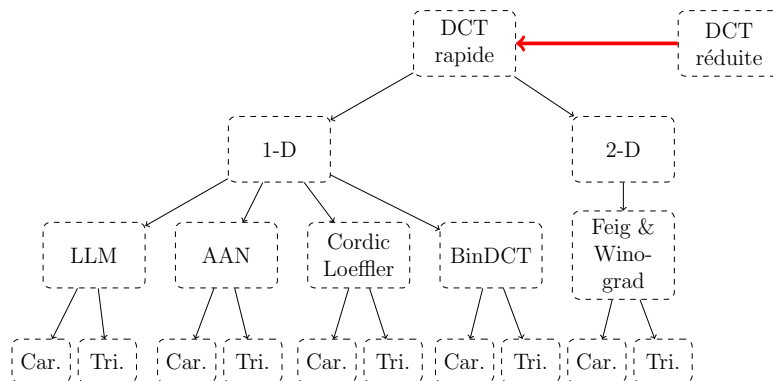
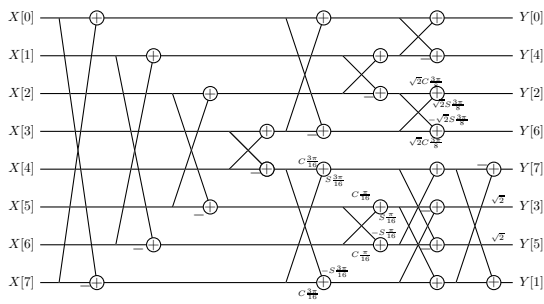


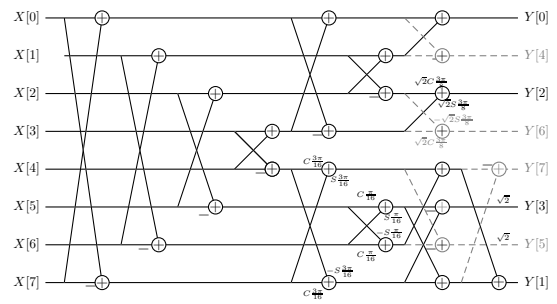
FIGURE 2.13: Les méthodes de calcul de la DCT zonale rapide

Nos contribution sont les suivantes :

- Nous appliquons la **DCT zonale** sur les algorithmes de DCT 1-D et 2-D comme Loeffler, Cordic Loeffler, BinDCT, Feig & Winograd, AAN (voir FIGURE 2.13).
- Nous proposons de **diminuer la complexité** des algorithmes de AAN et **Feig & Winograd** en remplaçant les multiplications par des additions et des décalages ;
- Nous **comparons deux modèles** de sélection zonale des coefficients de la DCT, l'un utilisant une zone de **forme carrée** et l'autre une zone de **forme triangulaire** ;
- Nous **évaluons la complexité de la DCT zonale rapide** en fonction de la



(a) Graphe de flux de Loeffler (8 entrées, 8 sorties)



(b) Graphe de flux de Loeffler (8 entrées, 4 sorties)

FIGURE 2.14: Graphe de flux de Loeffler

dimension de la zone de sélection des coefficients.

Le premier objectif de cette partie est de présenter les principaux algorithmes rapides pour le calcul des coefficients de la DCT (Loeffler, Cordic Loeffler, BinDCT, AAN, Feig & Winograd) appliquée sur des blocs de taille  $8 \times 8$ . Puis nous allons étudier les effets de la variation du paramètre de l'approche zonale  $k$  sur le coût de calcul des coefficients de la DCT, pour les zones des deux formes : carrée et triangulaire. Ensuite, nous allons proposer de remplacer les multiplications par des décalages et des sommations afin de réduire la complexité de l'algorithme. Nous allons enfin comparer les différentes méthodes, tout en tenant compte du compromis distorsion-complexité afin de retenir la meilleure méthode et forme.

### Cas de l'algorithme LLM (Loeffler-Ligtenberg-Moschytz)

Les auteurs dans (Loeffler et Moschytz, 1989) ont repris le même principe que Chen et ils ont proposé une méthode plus rapide qui permet d'optimiser la DCT-1D de Chen et calcule cette transformée avec seulement 11 multiplications et 29 additions (176 multiplications et 464 pour DCT 2-D). Il a été démontré que 11 multiplications est le nombre minimum théorique nécessaire pour le calcul de la transformation DCT-1D (Duhamel et H'Mida, 1987; Feig et Linzer, 1992). Le graphe des flux de l'algorithme LLM est présenté dans la FIGURE 2.14a. La DCT 2-D peut être obtenue en appliquant d'abord la DCT 1-D sur chaque ligne du bloc de pixels puis sur chacune des colonnes du bloc résultant.

Malgré le nombre réduit d'opérations, l'algorithme *LLM* reste gourmand en calcul. Afin d'adapter l'algorithme aux contraintes des réseaux de capteurs, nous avons présenté dans (Makkaoui *et al.*, 2010) une version zonale de l'algorithme LLM. Cela se traduit par une réduction du nombre de ses sorties à  $k$ .

Un exemple est montré dans Figure 2.14b pour le cas où  $k = 4$ .

Le tableau 2.3 donne le nombre d'opérations nécessaires pour calculer la DCT zonale rapide, en 1-D et en 2-D, pour les différentes valeurs de  $k$ .

Désignation	Coût de la DCT 1-D		Coût de la DCT 2-D (forme carrée)		Coût de la DCT 2-D (forme triangulaire)	
	Mult.	Add.	Mult.	Add.	Mult.	Add.
LLM(référence)	11	29	176	464	152	413
LLM Zonale (k=7)	11	28	165	420	130	348
LLM Zonale (k=6)	10	26	140	364	102	280
LLM Zonale (k=5)	9	25	117	325	77	223
LLM Zonale (k=4)	9	24	108	288	59	169
LLM Zonale (k=3)	8	23	88	253	38	118
LLM Zonale (k=2)	6	20	60	200	18	66

TABLE 2.3: Nombre d'opérations de l'algorithme Loeffler zonal

Comparé à l'algorithme LLM original, les économies de calcul de la DCT 1-D paraissent assez faibles (d'environ 18%). Cependant, lorsqu'on projette les résultats à la DCT 2-D, la réduction du coût de calcul devient significative puisque cela nous fait une économie d'environ 38% sur chaque type d'opération (pour  $k = 4$ ).

### Cas de l'algorithme AAN (Arai-Agui-Nakajima)

Cet algorithme a été développé par (ARAI *et al.*, 1988) et a été aussi étudié par (Kreyszig, 2007). AAN est considéré comme la DCT 1-D la plus rapide pour le calcul des coefficients qui doivent être quantifiés. Cet algorithme utilise 13 multiplications pour calculer la DCT 1-D mais parmi ces multiplications, 8 peuvent être incorporées dans la phase de quantification. Cela réduit le nombre d'opérations pour le calcul de l'algorithme AAN à 5 multiplications et 29 additions (80 multiplications et 464 additions pour le calcul de la DCT 2-D).

Afin de calculer la DCT 1-D, AAN utilise une factorisation de la matrice DCT en sous matrice comme suit :

$$\begin{pmatrix} F(0) \\ F(1) \\ F(2) \\ F(3) \\ F(4) \\ F(5) \\ F(6) \\ F(7) \end{pmatrix} = D_F \cdot P_F \cdot B_F \cdot M_F \cdot A_F \begin{pmatrix} f(0) \\ f(1) \\ f(2) \\ f(3) \\ f(4) \\ f(5) \\ f(6) \\ f(7) \end{pmatrix} \quad (2.17)$$



Les matrices  $D_F$ ,  $P_F$ ,  $B_F$ ,  $M_F$  et  $A_F$  sont définies par :

$$D_F = \begin{pmatrix} \frac{1}{2\sqrt{2}} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{4.C_1} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{4.C_2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{4.C_3} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2\sqrt{2}} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{4.C_5} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{4.C_6} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{4.C_7} \end{pmatrix} \quad (2.18)$$

$$P_F = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \quad (2.19)$$

$$B_F = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 \end{pmatrix} \quad (2.20)$$

$$M_F = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\cos \frac{\pi}{8} & 0 & -\cos \frac{3\pi}{8} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & 0 & 0 & -\cos \frac{3\pi}{8} & 0 & \cos \frac{\pi}{8} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (2.21)$$

Ainsi le graphe de flux de l'algorithme AAN, obtenue à partir des factorisations de la matrice transformée, est représenté dans la figure ci-dessous.

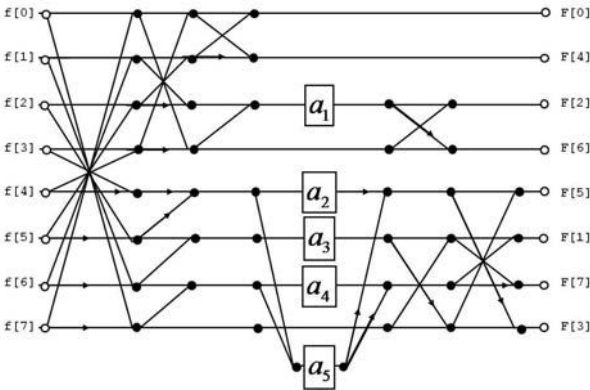


FIGURE 2.15: Graphe de flux de AAN (8 entrées, 8 sorties)

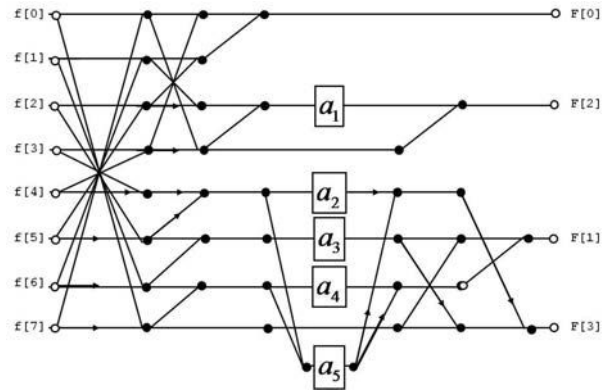


FIGURE 2.16: Graphe de flux de AAN  $k = 4$  (8 entrées, 4 sorties)

Les facteurs  $a_0, a_1, a_3, a_4$  et  $a_5$  sont définis comme suit :

$$\begin{aligned} a_1 &= a_3 = \cos \frac{\pi}{4} = \frac{1}{\sqrt{2}} \\ a_2 &= \cos \frac{\pi}{8} - \cos \frac{3\pi}{8} = \sqrt{1 - \frac{1}{\sqrt{2}}} \\ a_4 &= \cos \frac{\pi}{8} + \cos \frac{3\pi}{8} = \sqrt{1 + \frac{1}{\sqrt{2}}} \\ a_5 &= \cos \frac{3\pi}{8} = \frac{1}{2} \sqrt{2 - \sqrt{2}} \end{aligned}$$

Dans le tableau 2.4, nous présentons le nombre d'opérations de la DCT zonale basée sur l'algorithme.

Afin de diminuer le nombre de calcul, de nombreux travaux ont été réalisés pour définir une transformation DCT 2-D directe. Nous allons étudier dans le paragraphe suivant, l'algorithme proposé par (Feig et Winograd, 1992) qui permet le calcul direct de la DCT

	Coût de la DCT 1-D		Coût de la DCT 2-D (forme carrée)		Coût de la DCT 2-D (forme triangulaire)	
	Mult.	Add.	Mult.	Add.	Mult.	Add.
AAN (référence)	5	29	80	464	64	376
AAN Zonale (k=7)	5	28	75	420	54	318
AAN Zonale (k=6)	5	27	70	378	44	262
AAN Zonale (k=5)	5	26	65	338	34	208
AAN Zonale (k=4)	5	25	60	300	24	156
AAN Zonale (k=3)	4	23	44	253	14	106
AAN Zonale (k=2)	3	18	30	180	6	60

TABLE 2.4: Coût de calcul de l'algorithme AAN zonal

2-D.

### Cas de l'algorithme de Feig & Winograd

Nous trouvons dans la littérature de nombreuses recherches qui porte sur la proposition des algorithmes de calcul de la DCT 1-D qui peuvent être étendus pour le calcul des DCT 2-D. Mais nous ne trouvons pas beaucoup d'algorithmes qui permettent de calculer le DCT 2-D directement. Parmi les algorithmes efficaces dans le domaine de 2-D, l'algorithme proposé par (Feig et Winograd, 1992) est basé sur la méthode AAN (ARAI *et al.*, 1988).

De ce point de vue, Feig et Winograd ont proposé de calculer la DCT 2-D avec la multiplication tensorielle de la matrice 1-D DCT de AAN avec elle-même comme proposé dans les travaux de Tolimieri (Granata *et al.*, 1992).

Ce produit s'écrit sous la forme :

$$\begin{aligned} K_8 \otimes K_8 &= (D_F.P_F.M_F.A_F) \otimes (D_F.P_F.B_F.M_F.A_F) \\ &= (D_F \otimes D_F). [(P_F.B_F.M_F.A_F) \otimes (P_F.B_F.M_F.A_F)] \end{aligned}$$

Si on note  $K'_F = P_F.B_F.M_F.A_F$  on a alors :

$$K_8 \otimes K_8 = (D_F \otimes D_F).(K'_8 \otimes K'_8)$$

La méthode présentée par Feig et Winograd, est similaire à AAN en termes d'approximation de la DCT 2-D mais très efficace en termes de temps de calcul (54 multiplications et 462 additions).

En considérant la *DCT zonale rapide* basée sur l'algorithme de *Feig-Winograd*, on constate dans la table 2.5, que le nombre d'opérations de l'algorithme de *Feig-Winograd* est de 54 multiplications et 462 additions. Comparé à l'algorithme de *Loeffler* et *AAN*, la réduction du coût de calcul est significative puisque le gain est de 122 multiplications. Mais, en appliquant l'approche zonale le gain en nombre d'opération devient moins significatif puisque le coût de calcul de *Feig-Winograd* ne diminue pas beaucoup avec la diminu-

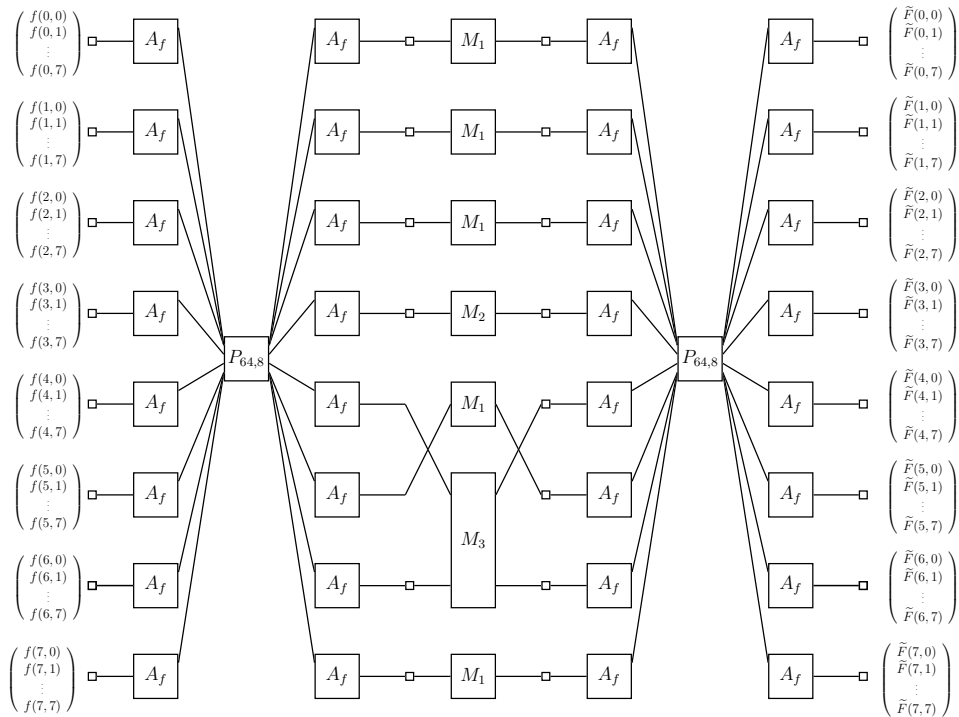


FIGURE 2.17: Graphe de flux de Feig-Winograd

tion de paramètre  $k$ . Les méthodes 1-D fournissent des performances qui se rapprochent, par exemple pour  $AAN$  avec  $k = 4$  le nombre d'opérations est de 60 multiplications et 300 additions contre 49 multiplications et 353 additions pour *Feig-Winograd* pour la même valeur de  $k$ . Les méthodes 1-D sont même meilleurs dans la forme triangulaire.

Bien que les approches calculant directement la DCT-2D ont une complexité de calcul moindre que les approches 1-D par lignes puis colonnes, elles exigent un espace mémoire important pour stocker les résultats intermédiaires. De plus, les nombreux échanges mémoire qui sont sollicités par le microcontrôleur peuvent dégrader la vitesse d'exécution.

	Coût de la DCT 2-D	
	Mult.	Add.
Feig & Winograd(référence)	54	462
Feig & Winograd Zonale (k=7)	54	439
Feig & Winograd Zonale (k=6)	54	418
Feig & Winograd Zonale (k=5)	54	399
Feig & Winograd Zonale (k=4)	49	353
Feig & Winograd Zonale (k=3)	38	309

TABLE 2.5: Coût de calcul de l'algorithme Feig &amp; Winograd

Les DCT utilisant des opérations sur des nombres réels ont un certain nombre d'inconvénients notamment liés à l'utilisation des multiplications qui sont coûteuses en énergie. Quelques algorithmes opèrent dans l'espace des valeurs entières et utilisent des additions et des décalages.

De nombreux travaux ont été réalisés afin de diminuer la complexité de la DCT 2-D réelle utilisée par JPEG parmi lesquelles les travaux de Trac D. Tran (Tran, 1999) (Tran, 2000).

Ces travaux ont été basés sur le fait que :

- Toute matrice orthogonale  $N \times N$  peut être factorisée en  $\frac{N(N-1)}{2}$  rotations.
- Toute matrice inversible  $N \times N$  peut être caractérisée par des facteurs diagonaux et des permutations.

### Cas de l'algorithme BinDCT

La factorisation de Chen permet de réaliser la transformation DCT 1-D en 13 multiplications et 29 additions. Toutes ces multiplications se situent au niveau des rotations des angles qui peuvent être représentées sous forme de décomposition de lifting (Bruekers et van den Enden, 1992). Dans ce cas, cette transformation n'aura besoin que d'additions et de décalages binaires. Elle n'utilise que des opérations sur des valeurs entières d'où son nom de BinDCT.

La structure générale de la BinDCT peut maintenant être déduite de la factorisation de Chen et des deux décompositions lifting des rotations selon 2.3.3

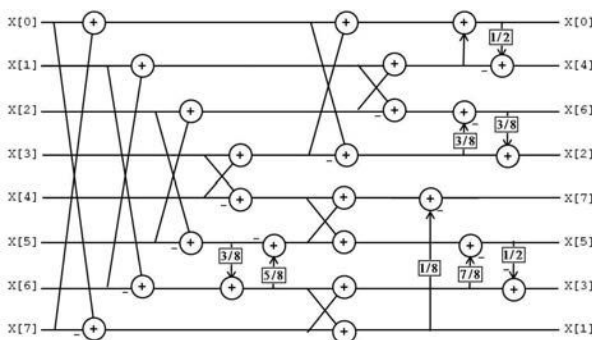


FIGURE 2.18: Graphe de flux de BinDCT (8 entrées, 8 sorties)

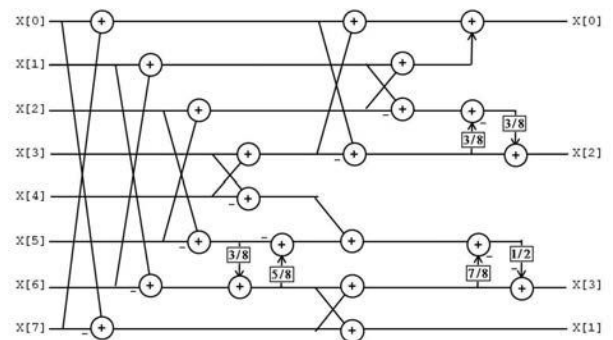


FIGURE 2.19: Graphe de flux de BinDCT  $k = 4$  (8 entrées, 4 sorties)

Les paramètres de lifting sont donnés en fonction de la configuration de la BinDCT. En effet, il existe différentes configurations  $C_i$  pour la BinDCT, chacune ayant une précision différente des approximations  $p$  et  $u$ . ces paramètres sont présentés en (Liang et Tran,

2000). Les paramètres du lifting sont donnés en fonction de la configuration de la BinDCT.

Pour vérifier les performances de la BinDCT, (Tran, 2000) a comparé deux types de BinDCT, celle basée sur l'algorithme de Chen et celle basée sur l'algorithme de Loeffler.

	DCT 1-D		DCT 2-D (forme carrée)		DCT 2-D (forme triangulaire)	
	Add.	Décal.	Add.	Décal.	Add.	Décal.
BinDCT C5 (référence)	36	19	576	304	482	240
BinDCT C5 Zonale (k=7)	36	19	540	285	410	202
BinDCT C5 Zonale (k=6)	36	19	504	266	338	164
BinDCT C5 Zonale (k=5)	36	19	468	247	266	126
BinDCT C5 Zonale (k=4)	35	18	420	216	194	88
BinDCT C5 Zonale (k=3)	31	15	341	165	124	52
BinDCT C5 Zonale (k=2)	24	11	240	110	62	22

TABLE 2.6: Coût de calcul de l'algorithme BinDCT zonal

De toutes ces méthodes approchées présentées par Tran, c'est l'algorithme de BinDCT Chen 5 (BinDCT C5), avec 36 additions et 19 décalages (soit 576 additions et 304 décalages en 2-D pour la forme carrée), qui fournit le meilleur compromis entre la complexité de calcul et la distorsion de l'image. Mais la BinDCT reste une approximation et elle ne fournit pas les mêmes précisions que les DCT réelles.

Les effets de la variation du paramètre  $k$  sur le coût de calcul des coefficients de la BinDCT, pour les formes carrée et triangulaire sont présentés dans le tableau 2.6. Les résultats montrent qu'on obtient des économies significatives en termes de calcul qui peuvent atteindre 60% pour  $k=2$ .

### Cas de l'algorithme CordicDCT (COordinate Rotation Digital Computer DCT)

L'algorithme CORDIC a été introduit à l'origine par (Volder, 1956) pour effectuer des rotations de vecteurs, il permet le calcul de racines carrées et des fonctions trigonométriques sinus, cosinus, tangente ainsi que leurs inverses.

Cet algorithme limite les opérations nécessaires à des additions, soustractions et décalages. Cette particularité a permis son utilisation pour le calcul de la DCT.

(Jeong *et al.*, 2004) ont été les premiers à proposer un algorithme qui permet d'améliorer le calcul de la DCT en se basant sur l'algorithme Cordic. Il consiste à calculer l'approximation de la DCT avec des rotations Cordic avec seulement 104 additions et 84 décalages. Il s'agit de remplacer les multiplications par des rotations d'angles.

Le vecteur  $(x', y')$  est défini comme le résultat de la rotation d'angle  $\theta$  du vecteur  $(x, y)$ .

Cette opération peut être décrite par l'équation 2.22 :

$$\begin{cases} x' = x \cdot \cos \theta - y \cdot \sin \theta \\ y' = x \cdot \sin \theta + y \cdot \cos \theta \end{cases} \quad (2.22)$$

Si on met  $\cos \theta$  en facteur et en se basant sur les relations trigonométriques  $\cos \theta = \cos(-\theta)$  et  $\tan \theta = -\tan(-\theta)$ , les équations dans 2.22 peuvent s'écrire :

$$\begin{cases} x' = \cos \theta (x - d \cdot y \cdot \tan \theta) \\ y' = \cos \theta (d \cdot x \cdot \tan \theta + y) \end{cases} \quad (2.23)$$

Mais il reste difficile à d'implanter ces équations efficacement à cause des multiplications par  $\cos \theta$  et  $\tan \theta$ . En effet, on cherche à les calculer.

La rotation d'angle  $\theta$  va donc être décomposée en une succession de rotations d'angles plus petits choisis de façon à être plus simples à calculer. On va utiliser des angles de telle sorte que :

$$\tan \theta_i = \pm 2^{-i} \quad (2.24)$$

Une rotation circulaire est décrite par l'équation 2.25.

$$\theta = \sum_i \sigma_i \cdot \tan^{-1} 2^{-i}, \sigma_i = \pm 1 \quad (2.25)$$

On peut introduire  $\sigma_i = \pm 1$  dans les équations pour contrôler le sens de rotation : c'est-à-dire, que l'angle de rotation sera  $\theta$  si  $\sigma_i = 1$  et  $-\theta$  si  $\sigma_i = -1$ . Ensuite, le calcul de la rotation d'un vecteur peut être effectué en faisant appel à une méthode itérative à l'aide de l'équation 2.26

Dans l'équation 2.26, les multiplications dans l'équation 2.23 seront remplacées par des simples opérations de décalage et d'additions.

$$\begin{cases} x_{i+1} = x_i - \sigma_i \cdot y_i \cdot 2^{-i} \\ y_{i+1} = y_i - \sigma_i \cdot x_i \cdot 2^{-i} \end{cases} \quad (2.26)$$

Ensuite, afin d'obtenir une approximation suffisamment précise, ces résultats doivent

être compensés comme montré dans l'équation 2.27.

$$\begin{cases} x_{i+1} = x_i \cdot (1 + \sigma_i \cdot F_i) \\ y_{i+1} = y_i \cdot (1 + \sigma_i \cdot F_i) \end{cases} \text{ avec, } \left\{ \prod (1 + \sigma_i \cdot F_i) \cong s \quad \text{et } \sigma_i = (0, 1, -1), F_i = 2^{-i} \right. \quad (2.27)$$

Donc pour réaliser une rotation d'un angle donné sur un vecteur, il faut lui appliquer une suite de n rotations successives dont l'angle à la ième itération est :

$$\tan^{-1}(2^{-i})$$

Le choix du nombre d'itérations dépend de la précision désirée.

Un résumé des rotations des angles utilisés est présenté dans la Table 2.7.

Ainsi la structure générale de la DCT peut être déduite à partir des rotations Cordic comme montré dans la FIGURE 2.20

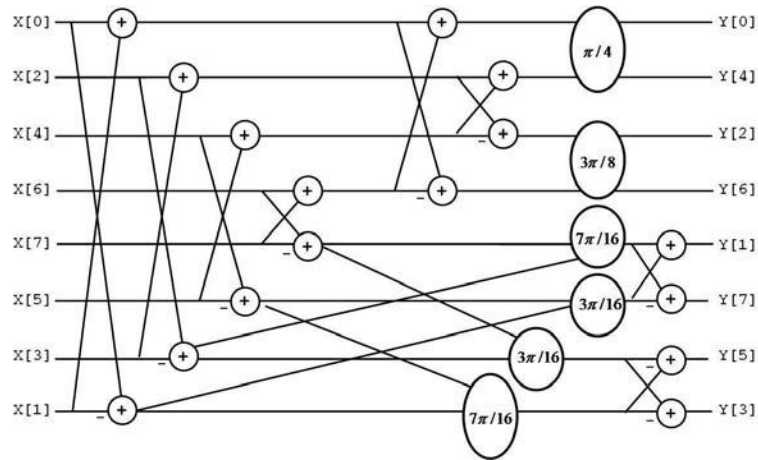


FIGURE 2.20: Graphe de flux de Cordic DCT

Pour diminuer le nombre d'opérations (104 additions et 82 décalages) de CordicDCT, (Sun *et al.*, 2007) ont combiné l'algorithme de Loeffler avec Cordic en remplaçant les multiplications par des rotations circulaires de Cordic.

En remplaçant les rotations dans le diagramme de Loeffler par des rotations Cordic, on obtient l'algorithme Cordic Loeffler, qui est présenté par le graphe suivant :



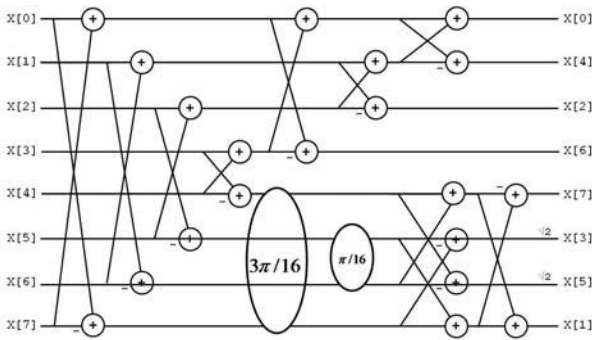


FIGURE 2.21: Graphe de flux de Cordic Loeffler(8 entrées, 8 sorties)

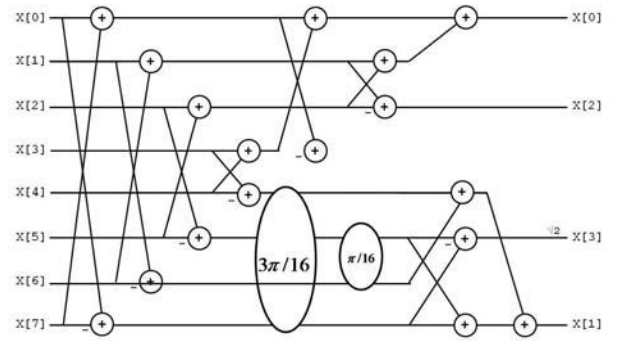


FIGURE 2.22: Graphe de flux de Cordic Loeffler (8 entrées, 4 sorties)

On constate que le nombre d'opérations est réduit à 38 additions et 19 décalages pour la DCT 1-D. Ce nombre reste légèrement plus grand que le nombre d'opérations de BinDCT mais la précision est meilleure.

Cordic	(1)	(2)	(3)	(4)
Angle	$\frac{\pi}{4}$	$\frac{3\pi}{8}$	$\frac{\pi}{16}$	$\frac{3\pi}{16}$
Les itérations Cordic $[\sigma, i]$				
1	-1, 0	-1, 0	-1, 3	-1, 1
2	-	-1, 1	-	-1, 4
3	-	+1, 4	-	-
Les itérations de compensations $[1 + \sigma_i \cdot F_i]$				
1	-	-	-	$1 - \frac{1}{8}$
2	-	-	-	$1 + \frac{1}{64}$

TABLE 2.7: Les paramètres Cordic pour les angles  $\frac{\pi}{4}$ ,  $\frac{3\pi}{8}$ ,  $\frac{\pi}{16}$  et  $\frac{3\pi}{16}$

La réduction du coût de calcul est significative puisque on gagne 368 additions et 146 additions ( $k = 2$ ). Cela fait une économie d'environ 60% sur le nombre d'additions. Le tableau 2.8 donne le nombre d'opérations nécessaires pour calculer la CordicDCT zonale rapide, en 1-D et en 2-D, pour les différentes valeurs de k.

	Coût de la DCT 1-D		Coût de la DCT 2-D zonale (forme carrée)	
	Add.	Décal.	Add.	Décal.
Cordic Loeffler(référence)	38	16	608	256
Cordic Loeffler (k=7)	37	16	555	240
Cordic Loeffler (k=6)	37	16	532	224
Cordic Loeffler (k=5)	35	15	455	195
Cordic Loeffler (k=4)	29	11	348	132
Cordic Loeffler (k=3)	28	11	308	121
Cordic Loeffler (k=2)	24	11	240	110

TABLE 2.8: Coût de calcul de l'algorithme Cordic Loeffler

### Algorithmes Feig & Winograd entier et intAAN proposés

Afin de diminuer la complexité et le temps de calcul de l'algorithme de Feig et Winograd décrit dans le paragraphe 2.3.3, nous nous sommes basés sur le fait que les opérations d'additions et de décalages sur nombres entiers sont beaucoup moins coûteuses en temps de calcul sur un processeur, qu'une multiplication sur des nombres réels. Nous avons remplacé les multiplications par des décalages et des additions (voir table 2.9) en se basant sur le fait que chaque nombre  $N$  peut être approximé par une sommation de puissance de 2 (Bracamonte *et al.*, 2006).

$$N = 2^{j_0} + 2^{j_1} + 2^{j_2} + \dots + 2^{j_m} \text{ Avec } m \geq 0 \text{ et } 0 \leq j_0 \leq j_1 \leq j_2 \dots \leq j_m$$

Coefficient original	Approximation sans multi.	Représentation binaire	Add.	Décal.	erreur
0,70710	0,703125	0.10110100	3	4	0,003
0,54119	0,53515625	0.10001001	2	3	0,006
0,38268	0,3828125	1.0101001	3	4	0,00013
1,30656	1,3046875	0.1010001	2	4	0,001

TABLE 2.9: Approximation des coefficients du calcul de la DCT

L'algorithme AAN permet de calculer la DCT 1-D avec seulement 5 multiplications. Mais comme nous pouvons encore diminuer le temps de calcul, nous avons procédé comme dans l'algorithme int Feig-Winograd, en remplaçant ces multiplications par des additions et des décalages.

	Coût de la DCT 1-D		Coût de la DCT 2-D zonale (forme carrée)		Coût de la DCT 2-D zonale (forme triangulaire)	
	Add.	Décal.	Add.	Décal.	Add.	Décal.
intAAN(référence)	19	42	304	672	254	434
intAAN (k=7)	19	41	285	615	216	450
intAAN (k=6)	19	40	266	560	178	368
intAAN (k=5)	19	39	247	507	140	288
intAAN (k=4)	19	38	228	456	102	210
intAAN (k=3)	16	34	176	374	64	134
intAAN (k=2)	12	30	120	300	32	66

TABLE 2.10: Coût de calcul de l'algorithme intAAN zonal

Le coût de calcul de l'algorithme que nous avons proposé est montré dans le tableau 2.10. Nous étudierons dans le paragraphe 2.3.4 l'effet d'approximer les multiplication par des additions et des décalages en terme du coût d'énergie.

### 2.3.4 Evaluation des performances

L'évaluation a été réalisée en considérant le microcontrôleur de la famille l'AMTEL ATmega128L et en observant le nombre de cycles que doit exécuter ce microcontrôleur pour appliquer la DCT 2-D sur un bloc de  $8 \times 8$  pixels. Le nombre de cycle était obtenu en utilisant l'outil logiciel WinAVR20100110, un outil de développement de la série d'Atmel AVR des microprocesseurs RISC.

Les résultats pour chaque méthode de DCT zonale rapide et pour chaque valeur de  $k$  sont donnés dans le tableau 2.11.

	k=8	k=7	k=6	k=5	k=4	k=3	k=2
Loeffler	732873	676494	588269	503150	485468	410264	339892
AAN	548181	498642	450356	404410	360037	310087	216439
Feig & Winograd	229437	221707	220019	219092	202684	151396	84894
BinDCT	16619	14727	12973	11420	11035	8546	6543
Cordic Loeffler	15679	14340	12927	11701	10495	9373	7521
AANint	15633	15308	12697	11328	9985	7978	5987
Feig & Winograd int	24354	23569	22045	21543	18953	17595	16695

TABLE 2.11: Nombre de cycles de la forme carrée des différents algorithmes pour le calcul de la DCT appliqués sur un bloc  $8 \times 8$  de l'image Lena obtenu par le simulateur AVR Studio avec le microcontrôleur ATmega128.

Dans les méthodes basées sur des nombres réels, l'algorithme Feig & Winograd reste le meilleur par rapport à Loeffler et AAN. Alors que dans les méthodes basées sur des nombres entiers, l'algorithme Cordic Loeffler est meilleur en termes de nombre de cycle.

	k=8	k=7	k=6	k=5	k=4	k=3	k=2
Loeffler	660676	601661	531818	463067	412858	348760	252234
AAN	487573	445936	403364	363262	321268	233764	181311
Feig & Winograd	215885	221707	220019	219092	202684	151396	84894
BinDCT	15519	13877	12324	10618	10130	7999	6182
Cordic Loeffler	14672	13127	12142	10980	9819	8734	7055
AANint	14665	13260	13082	10603	9306	7674	5555

TABLE 2.12: Nombre de cycles de la forme triangulaire des différents algorithmes pour le calcul de la DCT appliqués sur un bloc  $8 \times 8$  de l'image Lena obtenu par le simulateur AVR Studio avec le microcontrôleur ATmega128.

Nous avons aussi comparé les méthodes sur le critère de leur ratio énergie/distorsion. La distorsion de l'image compressée était basée sur la mesure du PSNR. Les résultats sont présentés en prenant l'exemple de l'image Lena encodée au débit de 0,5 bpp pour les deux formes de zones, carrée et triangulaire.

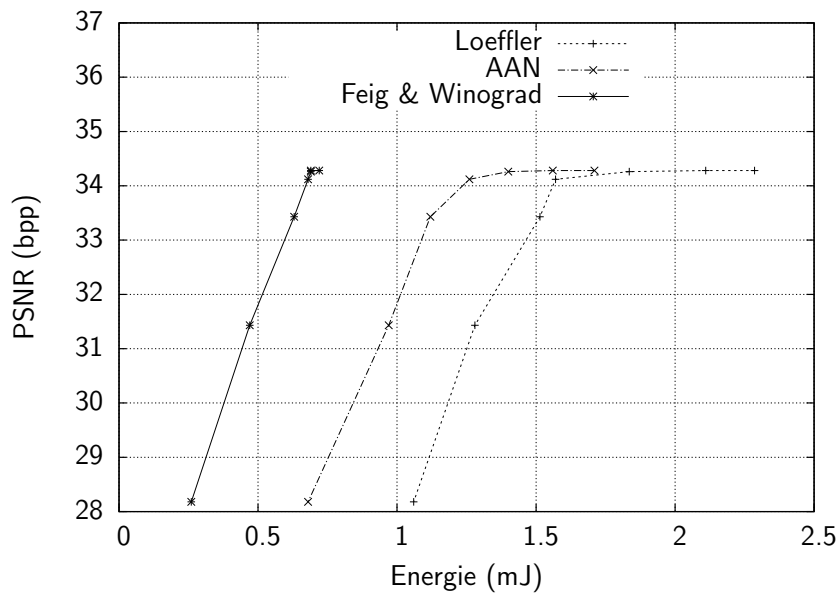


FIGURE 2.23: PSNR en fonction de l'énergie : Image Lena codée par le codage de Huffman à 0,5 *bpp* pour les DCT basées sur des nombres réels

Nous avons mené une série d'évaluations sur l'image Lena et avons étudié les effets de la variation du paramètre  $k$  sur le coût de calcul de toute la chaîne de compression, incluant la transformée DCT, la quantification et le codage entropique. Les résultats du PSNR en fonction du nombre de cycles sont présentés sur la FIGURE 2.23 (méthodes basées sur des opérations sur des nombres réels) et FIGURE 2.24 (méthodes basées sur des opérations sur des nombres entiers) et en considérant un débit de sortie de 0,5 *bpp*.

Nous pouvons vérifier que la forme triangulaire est plus économique en nombre de cycles puisqu'il y a moins de coefficients à calculer (TABLE 2.13), à quantifier et à encoder. En contrepartie, la distorsion de l'image sera plus élevée, pour des valeurs de  $k$  petites, puisqu'un plus grand nombre de coefficients sont négligés.

	Quantification scalaire carrée	Quantification scalaire triangulaire
k=8	57937	42721
k=7	49564	38168
k=6	42303	34179
k=5	36154	30754
k=4	31117	27893
k=3	27192	27192
k=2	24379	24379

TABLE 2.13: Nombre de cycles de la quantification scalaire pour un bloc ( $8 \times 8$ ) de l'image Lena

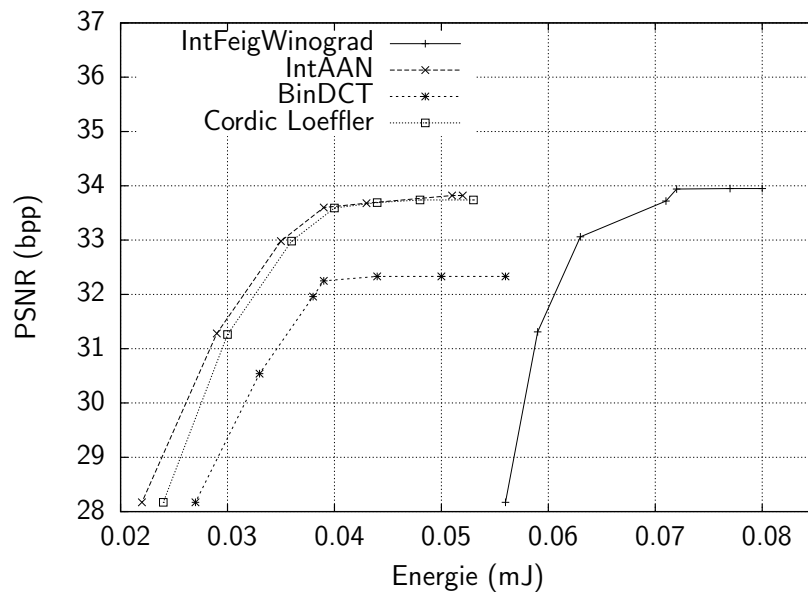


FIGURE 2.24: PSNR en fonction de l'énergie : Image Lena codée par le codage de Golomb+arithmétique à 0,5 *bpp* pour les DCT entières

Dans les méthodes utilisant une arithmétique sur des nombres réels, l'algorithme de Feig & Winograd présente le meilleur compromis énergie/distorsion par rapport aux algorithmes AAN et Loeffler comme montré dans la FIGURE 2.23. Pour les méthodes entières pour le calcul de la DCT, on constate clairement que les algorithmes de Cordic Loeffler et intAAN présentent des résultats similaires en termes de ratio énergie/distorsion, et nettement meilleurs que BinDCT et Feig & Winograd.

Bien que intAAN semble être légèrement plus performant que Cordic Loeffler, nous privilégions ce dernier car c'est un algorithme bien connu par la communauté de la compression d'images. intAAN exige plus d'opération que Cordic Loeffler (672 Add et 304 shifts contre 608 Adds et 256 shifts) et il nous semble rationnel de faire un choix par rapport à cela.

	Codage forme carrée	Codage forme triangulaire
k=8	546561	317288
k=7	426566	252167
k=6	308126	200350
k=5	231913	150074
k=4	156507	104414
k=3	111061	91244
k=2	73219	59601

TABLE 2.14: Nombre de cycles du codage de Huffman appliqués sur un bloc  $8 \times 8$  de l'image Lena (simulateur AVR Studio avec le microcontrôleur ATmega128)

	Codage forme carrée	Codage forme triangulaire
k=8	8740	5968
k=7	7585	5968
k=6	6516	5437
k=5	5737	4967
k=4	5044	4582
k=3	4505	4274
k=2	4120	4043

TABLE 2.15: Nombre de cycles du codage de Golomb+Arithmétique appliqué sur un bloc  $8 \times 8$  de l'image Lena (simulateur AVR Studio avec le microcontrôleur ATmega128)

Les tableaux 2.14 et 2.15 affichent les résultats obtenus en termes de nombre de cycles le codage de Golomb suivi par un codage arithmétique et pour le codage de Huffman. On constate que le nombre de cycles le plus élevé est obtenu par le codage de Huffman. On peut donc conclure que l'utilisation du codage de Golomb+arithmétique au lieu du codage de Huffman permet l'amélioration des performances de notre chaîne de compression.

## 2.4 Conclusion

Dans ce chapitre, nous avons d'abord décrit les principaux aspects de la compression et plus particulièrement la compression basée sur la DCT réduite. Cette DCT réduit le nombre de coefficients à calculer, et donc à quantifier et à encoder. Elle entraîne mécaniquement une réduction de la complexité de calcul de la chaîne de compression, et par incidence une réduction de la consommation d'énergie. D'après les résultats obtenus sur différentes images comme montré dans la TABLE 2.16, nous avons montré qu'une forme carrée fournit une meilleure efficacité énergétique qu'une forme triangulaire. En effet, avec une valeur d'énergie constante (dans notre cas égale  $0,04 \text{ mJ}$ ) le PSNR de la forme carrée est meilleur que celui de la forme triangulaire comme montré dans la TABLE 2.16.

	PSNR Carr. (dB)	PSNR Tri (dB)
Lena	33.34	33.28
Goldhill	30.65	30.30
Barbara	25.92	25.84
Elaine	31.93	31.99
Man	30.83	30.81

TABLE 2.16: PSNR avec la compression basée sur la méthode Cordic DCT avec énergie fixe= $0,04 \text{ mJ}$

Nous avons ensuite comparé le nombre de cycles des différents étapes d'une chaîne de compression <Transformée-Quantification-Codage>, afin de choisir la méthode la plus économique en calcul et en énergie. Nous avons déduit que la chaîne de compression <Cordic Loeffler zonale-Quantification scalaire-Codage Golomb+Arithmétique> présente un excellent compromis énergie/distorsion. L'approche zonale proposée, grâce à un paramètre  $k$  choisi, réduit le nombre de coefficients à calculer, à quantifier et à coder dans chaque bloc. Elle entraîne systématiquement une réduction de la complexité de calcul de la chaîne de compression, et par incidence une réduction de la consommation d'énergie et du temps d'exécution.

Nous avons étudié dans le chapitre suivant, le choix optimal du paramètre  $k$  qui permet de garantir une qualité visuelle acceptable de l'image reconstruite à un taux de compression donné.





# Chapitre 3

## DCT zonale rapide : choix de la taille de la zone en fonction du débit

### Sommaire

---

<b>3.1</b>	<b>Introduction</b>	<b>79</b>
<b>3.2</b>	<b>Les distorsions</b>	<b>80</b>
3.2.1	Les distorsions liées à la compression	80
3.2.2	Autres dégradations	81
<b>3.3</b>	<b>Mesures de la qualité des images reçues</b>	<b>81</b>
3.3.1	Les approches de type purement signal	82
3.3.2	Les approches structurelles	84
3.3.3	PSNR-HVS	88
<b>3.4</b>	<b>Expérimentation</b>	<b>88</b>
3.4.1	Bases d'évaluation	88
<b>3.5</b>	<b>Évaluation des performances</b>	<b>92</b>
<b>3.6</b>	<b>Conclusion</b>	<b>99</b>

---

### 3.1 Introduction

Depuis plusieurs années, de nombreuses recherches s'effectuent autour de l'estimation de la qualité d'une image (Eskicioglu et Fisher, 1995; Eskicioglu et Fisher, 1993). Les méthodes développées peuvent être classées en trois grandes catégories : celles qui sont à référence totale (utilisation complète de l'image originale), celles qui utilisent quelques paramètres de la référence, appelées aussi "méthodes à référence réduite" et celles qui n'utilisent aucune référence. Les méthodes "sans référence" se basent sur la connaissance a priori de la nature des dégradations ou des images. Les méthodes "sans référence" et "avec référence réduite" ne feront pas l'objet de cette thèse.

Dans ce chapitre nous cherchons à identifier le nombre de coefficients DCT à calculer  $k^2$  pour un débit donné sans perte significative de la qualité d'image. Pour cela, nous avons utilisé 3 métriques. La première métrique classique utilisée pour comparer deux images est le PSNR. Cette métrique tente de déterminer le niveau de distorsion d'une image compressée par rapport à sa source. Cette méthode reste purement mathématique et n'est pas basée sur le système visuel humain. La deuxième métrique utilisée est le SSIM (Structural SIMilarity) (Wang *et al.*, 2004) qui tente de déterminer la similarité structurelle des images, avec pour but d'être plus réaliste que le PSNR (plus fidèle au système visuel humain). Enfin nous utiliserons une troisième métrique, le PSNR-HVS (Egiazarian *et al.*, 2006; Ponomarenko *et al.*, 2007) qui est la mesure du PSNR pondéré par certaines propriétés du Système Visuel Humain.

## 3.2 Les distorsions

### 3.2.1 Les distorsions liées à la compression

Les distorsions introduites par notre chaîne de compression sont liées principalement à l'opération de quantification. D'autres facteurs affectent les distorsions produites, comme l'approximation des valeurs de la DCT et la réduction zonale de la DCT.

Différents types de distorsions peuvent être identifiés dans une image décompressée :

- L'effet de bloc : c'est l'apparition de motifs en forme de blocs dans l'image décompressée (de taille  $8 \times 8$  dans notre cas). L'effet de bloc est souvent la distorsion la plus frappante dans une image décompressée. Cet effet est dû à la différence de quantification d'un bloc à l'autre dans les schémas de compression utilisant une DCT par bloc, qui introduit des discontinuités à la frontière des blocs adjacents.
- Le flou : Il se manifeste comme une perte des détails, et comme une réduction de la finesse des contours. Il est dû à la suppression des coefficients hautes fréquences, après une quantification.
- Un effet "escalier" peut apparaître sur les contours obliques. Cet effet est dû au fait que les fonctions de base de la DCT sont plus adaptées à la représentation des contours horizontaux et verticaux. Les contours dont l'orientation n'est ni horizontale ni verticale nécessitent davantage de coefficients hautes fréquences pour obtenir une représentation précise. La quantification trop importante de ces coefficients introduit des irrégularités sur ces contours obliques.

### 3.2.2 Autres dégradations

A part les distorsions liées à la compression, la transmission des images peut être perturbée d'une part par des erreurs de transmission, d'autre part modifiée par l'existence de pré-traitements ou post-traitements. Les erreurs de transmission du flux binaire dans un canal de transmission bruité sont une source importante et souvent négligée de distorsions.

Deux différents types d'erreurs peuvent se produire pendant le transport sur des canaux bruités. Les paquets peuvent être perdus, ou ils peuvent être retardés suffisamment longtemps pour qu'ils ne soient pas reçus à temps pour le décodage. Le retard peut être lié aux techniques de routage ou de mauvais fonctionnement des capteurs.

Ces deux types d'erreurs ont le même effet : un ou plusieurs blocs de l'image ne sont pas disponibles lorsque des paquets sont manquants au moment où ils devraient être décodés. Les effets visuels de ces pertes varient beaucoup d'un décodeur à l'autre, en fonction de leur capacité à gérer des flux corrompus. Des décodeurs ne peuvent fonctionner en présence de certaines erreurs, tandis que d'autres utilisent des techniques adaptatives de dissimulation d'erreurs comme l'approximation de l'intensité des pixels non reçus à partir des pixels voisins.

## 3.3 Mesures de la qualité des images reçues

Comme nous l'avons déjà évoqué, deux grandes catégories de mesure de qualité peuvent être distinguées. La première utilise l'image seule pour mesurer la qualité (sans référence) et la seconde utilise une image de référence pour mesurer la qualité d'une image dégradée.

Dans la première catégorie, ces mesures sont orientées vers une dégradation spécifique, les mesures sans référence sont les plus attrayantes car elles permettent d'estimer la qualité uniquement à partir de l'image dégradée. Cependant, le type de dégradation contenu dans l'image est généralement supposé connu, ce qui limite fortement leur utilisation dans des situations pratiques.

Dans la seconde catégorie, ces types de mesure permettent de mesurer la différence entre une image originale et sa version dégradée. Généralement, aucune information liée au type de dégradation n'est exploitée. C'est cette seconde catégorie à laquelle nous nous intéressons plus particulièrement.

Il existe une grande variété de métriques de qualité d'images numériques avec référence. Les travaux de Pederson (Pedersen et Hardeberg, 2009) font état de plus de 111 métriques existantes. Depuis cette étude, une dizaine de métriques peuvent être ajoutées et ceci

est dû à l'effervescence que connaît cette problématique. On peut les classer en trois grandes catégories : mesures mathématiques, mesures pondérées par quelques propriétés du Système Visuel Humain (SVH) et la modélisation complète du SVH.

### 3.3.1 Les approches de type purement signal

Les méthodes les plus directes pour mesurer la distorsion sont les méthodes purement mathématique basées sur une différence mathématique  $D(m, n)$  pixel à pixel entre l'image originale  $I_o(m, n)$  et l'image dont la qualité est à évaluer  $I_d(m, n)$  :

$$D(m, n) = |I_o(m, n) - I_d(m, n)|^p \quad (3.1)$$

Lorsque  $p = 2$ , cette égalité permet de calculer l'erreur quadratique moyenne (MSE, Mean square error), qui est définie par l'équation 3.2.

$$MSE = \frac{1}{M.N} \sum_{m=1}^M \sum_{n=1}^N \|I_o(m, n) - I_d(m, n)\|^2 \quad (3.2)$$

La métrique de type signal la plus connue et la plus utilisée en image est le Peak Signal to Noise Ratio (PSNR) qui est corrélé à la MSE. Le PSNR est défini par l'équation 3.3.

$$PSNR = 10 \cdot \log_{10} \left( \frac{d^2}{MSE} \right) \quad (3.3)$$

où  $d$  est l'amplitude maximale du signal. En général, dans le cas d'une image où les composantes d'un pixel sont codées sur 8 bits,  $d = 255$ .

L'avantage de cette mesure mathématique réside dans sa simplicité d'implantation, ainsi que dans la rapidité de calcul. Par contre, elle présente l'inconvénient majeur de n'être basée que sur le signal et montre une faible corrélation avec le jugement humain. Elle ne prend donc pas en compte les propriétés du système visuel humain sur certaines images. Pour s'en convaincre, il suffit de regarder l'exemple de la FIGURE 3.1, dans lequel une image originale est présentée avec une version dégradée, ainsi que la carte d'erreurs mathématique associée. L'interprétation de cette carte laisserait penser que les distorsions dans la zone correspondant au bateau sont plus visibles que dans la zone correspondant au ciel, ce qui n'est pas le cas.

Les approches de type purement signal ne modélisent pas le système visuel humain, elles font l'hypothèse que la qualité visuelle décroît quand la distorsion du signal augmente. Cependant, la qualité ne dépend pas seulement des distorsions mais aussi de nombreux autres paramètres comme le contenu de l'image, la sensibilité au contraste, ou encore de la

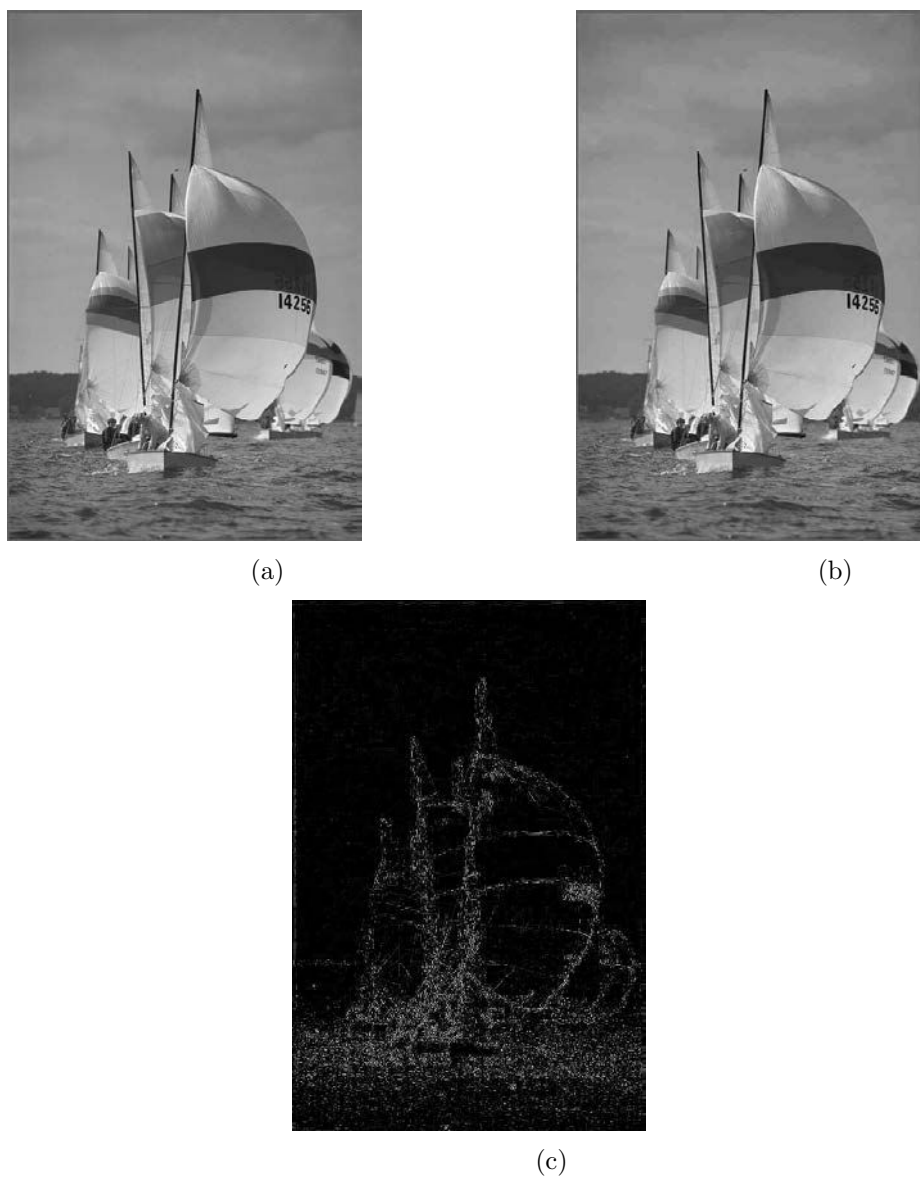


FIGURE 3.1: Carte d'erreurs mathématiques (c) entre l'image bateau originale (a) et une version compressé (b) codée à  $0,5 \text{ bpp}$ . Plus les valeurs sont sombres, et plus l'erreur mathématique est faible.

localisation des dégradations. La FIGURE 3.2 illustre les mauvaises performances du PSNR sur deux images. Nous pouvons constater que l'image Lena est notée moins sévèrement par le PSNR que l'image Mandrill, alors qu'il apparait clairement, en regardant ces deux images, que l'image Mandrill est de meilleure qualité visuelle. Une explication vient de ce que l'image Mandrill contient beaucoup de structures fines ayant un potentiel de masquage important. Les erreurs sont donc faiblement perçues.

Cette mesure, même si elle est utilisée très fréquemment en traitement d'image pour évaluer la qualité, ne permet pas de prendre en compte la sensibilité du système visuel humain, et n'apparait pas très performante. Ce qui a poussé à de nouvelles recherches pour mettre en place des nouvelles approches qui pallient à ce défaut.



FIGURE 3.2: Comparaison de la qualité visuelle de (a) l'image Lena  $PSNR = 24.84$  dB et (b) l'image Mandrill  $PSNR = 24.78$  dB

### 3.3.2 Les approches structurelles

Les approches structurelles tirent parti des avantages des critères du SVH et de l'utilisation des mathématiques pour automatiser et simplifier les calculs. Cependant, ces mesures sont généralement difficiles à concevoir de part la complexité du système visuel humain.

Parmi les mesures basées sur l'approche structurelle figure celle proposée par Z. Wang et A. C. Bovik dans leurs articles (Wang et Bovik, 2002; Wang *et al.*, 2003; Wang *et al.*, 2004; Wang *et al.*, 2005). Ils présentent un indice pour la qualité d'image.

Wang et. al définit cette nouvelle mesure de la manière suivante : "La fonction principale du système visuel humain est d'extraire l'information structurelle du champs d'observation et le système visuel humain y est fortement adapté. Donc une mesure de distorsion structurelle devrait être une bonne approximation de la distorsion perçue".

L'idée principale de l'approche proposée est de mesurer la "similarité de structure" ( $s(x, y)$ ) entre deux images, plutôt qu'une différence pixel à pixel comme le font les approches purement de type signal.

Toutefois, le calcul de similarité ne se limite pas seulement à la comparaison des structures entre les images, les différences de luminance<sup>1</sup> ( $l(x, y)$ ) et de contraste entre les deux images sont également évaluées<sup>2</sup> ( $c(x, y)$ ). Puisque ces derniers jouent effectivement un rôle important dans la perception, Wang développe la métrique SSIM pour "Structural Similarity Index Measurement" dont la formule est celle de l'équation 3.4.

Le calcul de similarité s'effectue donc en combinant trois mesures : une mesure de similarité de luminance  $l(x, y)$ , une mesure de similarité de contraste  $c(x, y)$  et une mesure de similarité de structure  $s(x, y)$ . La mesure entre deux fenêtres de taille  $N \times N$ , avec deux images :  $x = \{x_i | i = 1, 2, \dots, N\}$  et  $y = \{y_i | i = 1, 2, \dots, N\}$  est :

$$SSIM(x, y) = [l(x, y)]^\alpha \cdot [c(x, y)]^\beta \cdot [s(x, y)]^\gamma \quad (3.4)$$

avec :

- La distorsion de la luminance qui mesure la ressemblance de luminance entre  $x$  et  $y$  :

$$l(x, y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \quad (3.5)$$

- La distorsion du contraste qui mesure les similarités de contraste entre  $x$  et  $y$  :

$$c(x, y) = \frac{2\sigma_x\sigma_y}{\sigma_x^2 + \sigma_y^2 + C_2} \quad (3.6)$$

- La perte de corrélation qui mesure le degré de corrélation linéaire entre  $x$  et  $y$  :

$$s(x, y) = \frac{\sigma_{xy} + C_3}{\sigma_x \cdot \sigma_y + C_3} = \frac{cov(x, y) + C_3}{\sigma_x \cdot \sigma_y + C_3} \quad (3.7)$$

---

1. La luminance est la mesure de l'intensité lumineuse visible d'une source donnée  
 2. Le contraste est une propriété intrinsèque à une image qui permet de quantifier la capacité à distinguer deux régions distinctes.

Les variables  $C_1$  et  $C_2$  sont rajoutées pour résoudre le problème de la moyenne ou de l'écart-type nul. Elles valent  $C_1 = (K_1 \times L)^2$  et  $C_2 = (K_2 \times L)^2$ .  $L$  représente la dynamique du pixel (256 pour une image à niveaux de gris codée sur 8 bits),  $K_1$  et  $K_2$  des constantes de valeur 0,01 et 0,03.

Avec :

- La luminance de chaque image (moyenne de l'intensité) :

$$\mu_x = \frac{1}{N} \sum_{i=1}^N x_i, \quad \mu_y = \frac{1}{N} \sum_{i=1}^N y_i \quad (3.8)$$

- Le contraste de chaque image :

$$\sigma_x^2 = \frac{1}{N-1} \sum_{i=1}^N (x_i - \mu_x)^2, \quad \sigma_y^2 = \frac{1}{N-1} \sum_{i=1}^N (y_i - \mu_y)^2 \quad (3.9)$$

- Et

$$\sigma_{xy} = \frac{1}{N-1} \sum_{i=1}^N (x_i - \mu_x)(y_i - \mu_y) \quad (3.10)$$

En appliquant le SSIM sur une fenêtre glissante centrée sur chaque pixel  $(m, n)$  des images à comparer, il est possible de créer une carte des erreurs structurelles. La FIGURE 3.3 présente la carte SSIM calculée entre les images (a) et (b) de la FIGURE 3.1. On peut observer sur cette carte que les contours à fort contraste, comme les contours du bateau, ont un effet perturbateur sur les valeurs de SSIM calculées dans les fenêtres les contenant. Le fort contraste du contour entraîne une trop forte corrélation entre les deux fenêtres, ce qui provoque une sous estimation des erreurs par le SSIM. Par contre, on peut observer la détection des frontières des effets de blocs dans les zones de ciel, ce qui est un point intéressant. Cependant, comme pour la carte des erreurs mathématiques de la FIGURE 3.1, les valeurs relatives des erreurs mesurées entre les zones de ciel et de la mer laissent penser que le ciel est visuellement plus dégradé que la mer.

Dans la continuité du SSIM, d'autres méthodes basées sur les erreurs structurelles ont été proposées. On peut citer le SSIM multi-échelle (MS-SSIM multi-scale SSIM) également proposé par Wang et al. dans (Wang *et al.*, 2003). Cette méthode reprend les concepts de SSIM mais les applique à une approche multi-résolution.

Les niveaux de résolutions sont calculés à partir des images de départ par filtrage passe-bas et sous-échantillonnage. Les mesures  $l(m, n)$ ,  $c(m, n)$  et  $s(m, n)$  sont calculées



à différentes résolutions puis combinées selon la relation dans l'équation 3.11.

$$MS - SSIM(I_o, I_c) = [l(I_o, I_c)]^{\alpha_M} \cdot \prod_{j=1}^M [c(I_o, I_c)]^{\beta_j} \cdot [s(I_o, I_c)]^{\gamma_j} \quad (3.11)$$

$j$  représentant les différents niveaux de résolutions,  $M$  étant le nombre de niveaux de résolution utilisé, et les paramètres  $\alpha_M$ ,  $\beta_j$  et  $\gamma_j$  permettant d'ajuster l'importance des différentes composantes.

Cette approche est plus flexible que le SSIM, et tente d'inclure des notions de variations de la sensibilité à travers des différents niveaux de résolution.

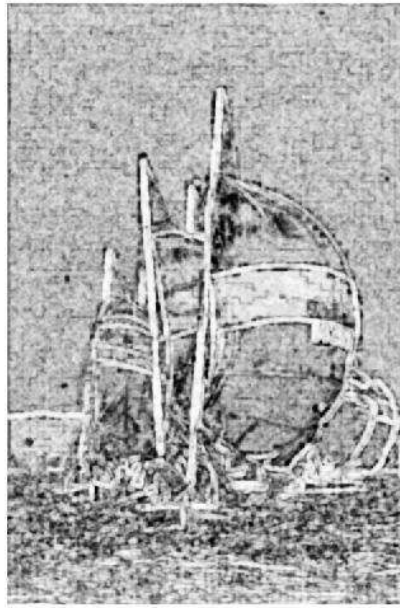


FIGURE 3.3: Carte d'erreurs structurales SSIM, entre l'image Bateau originale (FIGURE 3.1(a)), et une version dégradée (FIGURE 3.1(b)). Plus les valeurs sont claires, et plus les erreurs structurales sont faible.

Pour mesurer la qualité de l'image par l'utilisation de l'indice SSIM, on a besoin d'une seule mesure totale de la qualité de toute l'image. D'où un indice SSIM moyen (MSSIM) pour évaluer la qualité totale de l'image. Cet indice est déterminé par l'équation 3.12. La valeur est comprise entre 0 et 1, 1 étant la qualité la plus élevée.

$$MSSIM(I_o, I_c) = \frac{1}{M} \sum_{i=1}^M SSIM(x_i, y_i) \quad (3.12)$$

– Où  $I_o$  et  $I_c$  sont respectivement les images de référence et compressée.

- $x_i$  et  $y_i$  sont les contenus des images de la  $i$ ème fenêtre locale.
- $M$  : le nombre total de fenêtre locales dans l'image.

Notre expérience de cette métrique montre quelle améliore significativement l'évaluation de la qualité. Cependant, elle a tendance à ne pas faire la différence entre un contour et une texture puisqu'elle utilise la variance du bloc.

La plupart des métriques basées sur l'approche structurale sont construites sur l'approche *SSIM* quand elles incorporent des propriétés du système visuel humain.

### 3.3.3 PSNR-HVS

La métrique PSNR-HVS (Ponomarenko *et al.*, 2007), version modifiée du PSNR pondérée par certaines propriétés du Système Visuel Humain (PSNR-HVS) est connue par sa forte corrélation avec le MOS (Mean Opinion Score)<sup>3</sup>. Sa formule est donnée ci-dessous :

$$PSNR - HVS = 10 \log \left( \frac{255^2}{MSE_H} \right) \quad (3.13)$$

Dans l'équation 3.13,  $MSE_H$  est calculé en se basant sur les caractéristiques du système visuel humain :

$$MSE_H = K \sum_{i=1}^{I-7} \sum_{j=1}^{J-7} \sum_{m=1}^8 \sum_{n=1}^8 ((X[m, n]_{ij} - X'[m, n]_{ij})T_c[m, n])^2 \quad (3.14)$$

où :  $M, N$  représentent respectivement le nombre de lignes et le nombre de colonnes de l'image,  $K = \frac{1}{[(I-7)(J-7)64]}$ ,  $X_{ij}$  sont les coefficients de la DCT du bloc  $8 \times 8$  de l'image qui a pour coordonnées  $i$  et  $j$ .  $X'_{ij}$  sont les coefficients de la DCT du bloc de l'image originale, et  $T_c$  est la matrice de coefficients de correction proposée dans (Ponomarenko *et al.*, 2007).

La somme des facteurs de correction de la matrice  $T_c : \sum_{m=1}^8 \sum_{n=1}^8 \frac{T_c[m, n]^2}{64}$  est égale à 1. Cela permet de garantir, en cas d'une distribution uniforme des distorsions entre les fréquences, que la valeur de  $MSE_H$  coïncide avec la valeur de MSE.

## 3.4 Expérimentation

### 3.4.1 Bases d'évaluation

Afin d'évaluer les performances de l'approche de compression proposée en termes de qualité d'image, nous avons utilisé une base d'images composée de 60 images de référence

---

3. Le score d'opinion moyen (MOS) consiste à regrouper les résultats des observateurs humain pour obtenir une note moyenne sur une image. C'est-à-dire que chaque image a un score correspondant à la moyenne de toutes les notes qui lui ont été attribuées.

Taille de l'image (pixels)	128 × 128	200 × 200	256 × 256	512 × 384	512 × 512	640 × 512	480 × 720	768 × 512	1024 × 1024
Nombre d'images	2	1	1	11	17	6	6	15	1

TABLE 3.1: Description de la base d'images de test

de différentes tailles (voir table 3.1).

### Base IVC

Cette base d'images que nous appellerons base IVC, est issue des travaux de Le Callet (Le Callet, 2001). Elle a été élaborée à partir de huit images de scènes naturelles. Ces images, illustrées FIGURE 3.4, sont de taille  $512 \times 512$  pixels et ont été sélectionnées parmi celles communément utilisées par les concepteurs de méthodes de compression d'images. Pour chacune d'elles, des versions plus ou moins dégradées ont été générées en utilisant la chaîne de compression  $\langle$  CordicDCT, Quantification scalaire, Codage Golomb+arithmétique  $\rangle$ .

La variation du paramètre  $k$  et l'utilisation de plusieurs taux de compression conduisent à un total de 126 images dégradées.

### Base Tayoma

La seconde base de données d'évaluations subjectives, que nous appellerons base Toyama, est issue des travaux de Sazzad et al. (Sazzad *et al.*, 2007) de l'Université de Toyama<sup>4</sup> au Japon. Nous avons choisi vingt six images de cette base. Parmi ces images, illustrées sur FIGURE 3.5, quinze sont de taille  $768 \times 512$  pixels et onze de taille  $512 \times 384$ . Pour chacune d'elles, des versions plus ou moins dégradées ont été générées en utilisant notre système de compression décrit dans le chapitre 2.

Ce système permet d'obtenir des distorsions de natures différentes par la diminution de coefficients DCT à calculer  $k^2$  par bloc  $8 \times 8$ . L'utilisation de plusieurs taux de compression, avec le paramètre  $k$  allant de 2 à 8, permet d'obtenir un total de 198 images dégradées.

### Autres images

Nous avons complété les images obtenues de la base de test IVC et Tayoma par 26 images. Ces images sont de divers tailles mais elle sont toutes codées à 8 *bpp*.

---

4. Graduate School of Engineering, University of Toyama



FIGURE 3.4: Images originales à niveau de gris de la base de test IVC

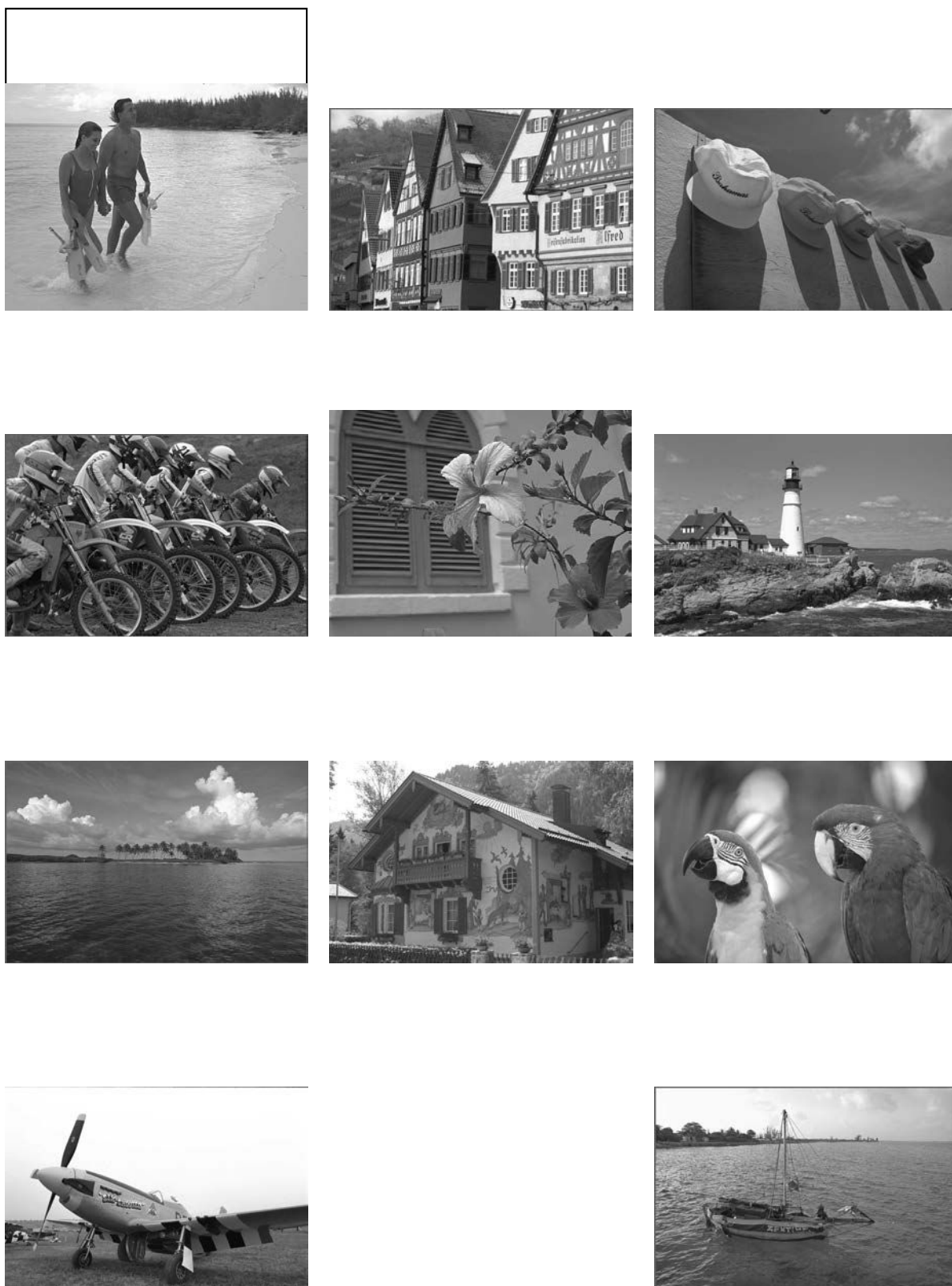


FIGURE 3.5: Images originales transformées à niveau de gris de la base de test Tayoma

### 3.5 Évaluation des performances

Dans cette section, nous allons évaluer les performances de notre chaîne de compression en proposant le choix de la valeur de  $k$  optimal. Le problème qui se pose est dans quelle mesure est-il possible de diminuer le paramètre  $k$  sans dégrader la qualité visuelle des images reconstruites ?

L'étude de la meilleure valeur de  $k$  nécessite la considération des trois points importants suivants :

- Le débit binaire de compression, car plus l'image est compressée, plus les composantes dans un bloc deviennent non significatifs (égaux à zéro), donc  $k$  peut être plus petit sans effet négatif sur la qualité de l'image.
- Les caractéristiques de l'image (luminance, texture...etc.)
- La façon avec laquelle la qualité visuelle de l'image est évaluée.

Concernant le premier point, la détermination du paramètre  $k$  a été faite en fonction du débit binaire de compression. Pour le second point, nous avons utilisé une base de 60 images de référence, présentée dans la section précédente. Les résultats obtenus sont moyennés pour chaque débit fixe. Concernant le troisième point, trois métriques pour l'évaluation de la qualité des images ont été considérées : le rapport signal sur bruit (PSNR), l'indice multi-échelle de similarité structurelle (MS-SSIM) et PSNR-HVS (PSNR basé sur le système visuel humain) déjà présentés dans la section 3.3 .

Comme nous l'avons déjà suggéré dans l'introduction de ce chapitre, l'évaluation de la qualité d'une image compressée avec pertes n'est pas une tâche facile. L'objet de ce paragraphe est de comparer les résultats d'outils parmi les plus utilisés dans ce domaine.

L'utilisation des outils basés sur une mesure de distorsion tel que le PSNR donnent une appréciation de la qualité d'ensemble de l'image et elles ne permettent pas de localiser une erreur dans l'image (Ninassi, 2009). Par ailleurs, elle est simple d'utilisation.

Nous avons alors choisi d'ajouter des simulations en utilisant d'autres méthodes qui se basent sur le système visuel humain (PSNR-HVS et SSIM).

Les FIGURE 3.6, FIGURE 3.8 et FIGURE 3.9 représentent un exemple des résultats obtenus par les métriques PSNR, PSNR-HVS et SSIM sur respectivement l'image Lena, Pappers et Barbara codées à 0,5 bpp.

Comme on peut le constater sur ces figures, pour des grandes valeurs de  $k$  (de 4 à 8) les allures des courbes de PSNR, PSNR-HVS et SSIM sont très proches. Pour les valeur de  $k = 2$  et  $k = 3$ , les résultats de PSNR et PSNR-HVS montrent une dégradation importante de la qualité de l'image par rapport à  $k = 4$ . Mais nous remarquons qu'il

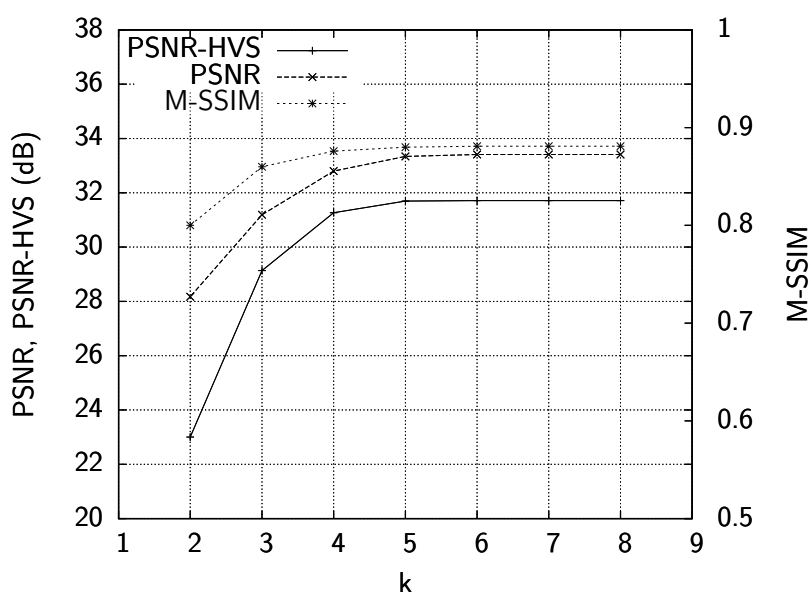


FIGURE 3.6: PSNR, PSNR-HVS et SSIM de l'image Lena codée à 0,5 bpp

n'y a pas une baisse importante de l'indice du SSIM pour ces mêmes valeurs de  $k$ . Cela s'explique d'une part par le fait que pour les indices SSIM, le choix de la taille de la fenêtre n'est pas satisfaisante. Si une petite fenêtre est choisie, les valeurs de variances ou de moyennes (à la base du SSIM) ne signifient que peu de choses. Si une grande fenêtre est retenue, les fines structures ne sont pas observables. D'autre part, pour des débit binaires bas, les indices SSIM ne reflètent pas le flou dans l'image.

Nous avons utilisé l'image Lena de taille  $512 \times 512$  pixels en niveaux de gris compressée à 0,5 bpp. En appliquant l'algorithme de compression JPEG, le PSNR vaut 33,41 dB, le PSNR-HVS=31,71 dB et le MS-SSIM est de l'ordre de 0,881. Quand l'approche zonale est utilisée, le choix de la valeur  $k = 5$  est raisonnable (PSNR = 33,34 dB, PSNR-HVS=31,69 dB et MS-SSIM = 0,880) car la qualité visuelle est similaire, comme indiquée dans la FIGURE 3.10.

Nous avons considéré une autre image qui est la célèbre Barbara en niveaux de gris et compressée à 0,5 bpp. L'apparition des effets de bloc est observé au niveau de l'image avec  $k = 5$  (voir FIGURE 3.7(c)).

En répétant les mêmes expériences sur l'ensemble des 60 images de référence et en moyennant les résultats, nous avons cherché à déterminer la valeur optimale statistique de  $k$  pour un débit binaire donné.

Nous avons imposé la condition, pour déterminer la bonne valeur de  $k$ , que les valeurs de PSNR, PSNR-HVS et MS-SSIM diffèrent simultanément de moins de 1% avec celles



(a) Barbara Original

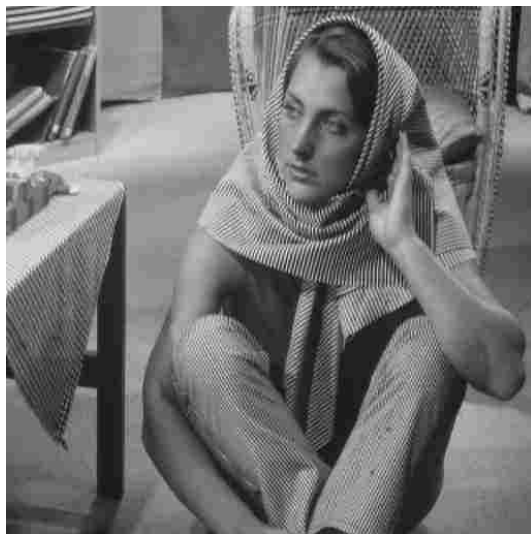
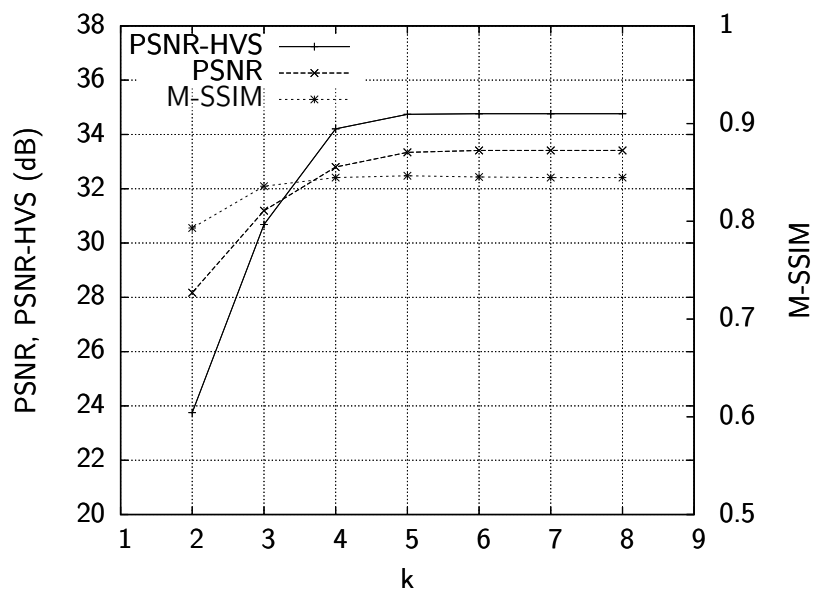
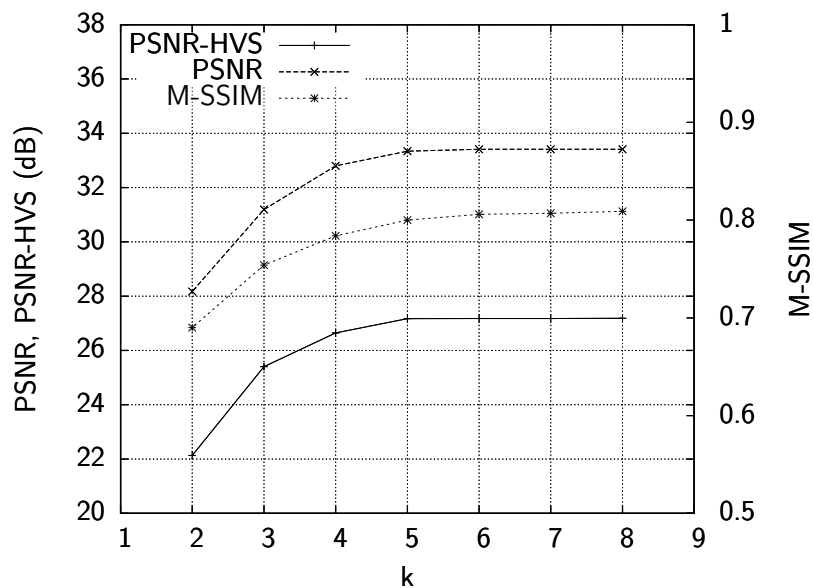
(b)  $PSNR = 26,84$  dB, MS-SSIM=0,811(c)  $k=5$ ,  $PSNR = 23,05$  dB, PSNR-HVS=27,17, MS-SSIM=0.8

FIGURE 3.7: Comparaison de la qualité visuelle de l'image Barbara compressée par JPEG à  $0,5$  *bpp* (b) et l'image Barbara codée par notre chaîne de compression (c)



FIGURE 3.8: PSNR, PSNR-HVS et SSIM de l'image Peppers codée à 0,5 *bpp*FIGURE 3.9: PSNR, PSNR-HVS et SSIM de l'image Barbara codée à 0,5 *bpp*

observées avec  $k = 8$ , afin de garantir aucune dégradation significative de la qualité visuelle de l'image.

Selon nos expériences, cette différence de 1% est suffisante, sans perdre beaucoup en qualité de l'image, pour évaluer notre système de compression.

Prenons par exemple, l'image Lena en niveaux de gris et compressée à 0,5 bpp. Quand l'algorithme JPEG est utilisé comme codeur (par exemple,  $k = 8$ , donc 64 coefficients DCT par bloc  $8 \times 8$  sont calculés, quantifiés et codés), le PSNR est égal à 33,56 dB et le MS-SSIM est de l'ordre de 0,883. Quand notre chaîne de compression est adoptée comme codeur, le choix de la valeur  $k = 5$  est raisonnable ( $PSNR = 33,34$  dB et  $MS-SSIM = 0,880$ ) car la qualité visuelle est similaire, comme indiquée dans la FIGURE 3.10(b).



(a)  $PSNR = 33,56$  dB,  $MS-SSIM = 0,883$



(b)  $k = 5$ ,  $PSNR = 33,34$  dB,  $MS-SSIM = 0,880$

FIGURE 3.10: Comparaison de la qualité visuelle de (a) l'image Lena compressée par JPEG à 0,5 bpp et (b) l'image Lena codée par notre chaîne de compression

On considère maintenant l'image Barbara en niveaux de gris et compressée à 0,5 bpp. Cette fois, nous n'avons aucune perte au niveau de la qualité visuelle de l'image avec  $k = 6$  (voir FIGURE 3.7(c)). En répétant les mêmes expériences sur notre base d'images composée de 60 images de référence et en moyennant les résultats, nous déterminons la valeur optimale statistique de  $k$  pour un débit binaire donné.

Les expériences ont été faites dans un intervalle de débit binaire allant de 0,15 à 1,0 bpp par pas de 0,05. Le choix cet intervalle de débit binaire se justifie par nos constatations à partir des résultats obtenus : à 1 bpp les images ont une très bonne qualité et au dessous de 0,15 la qualité est trop dégradée pour être utile à l'utilisateur final.

Les courbes donnant la valeur moyenne de  $k$  en fonction du débit binaire sont présentées dans les figures 3.11, 3.12 et 3.13.

Avec les métriques utilisés, les images sont évaluées à différentes résolutions et le résultat est la moyenne des résultats des étapes calibrées.

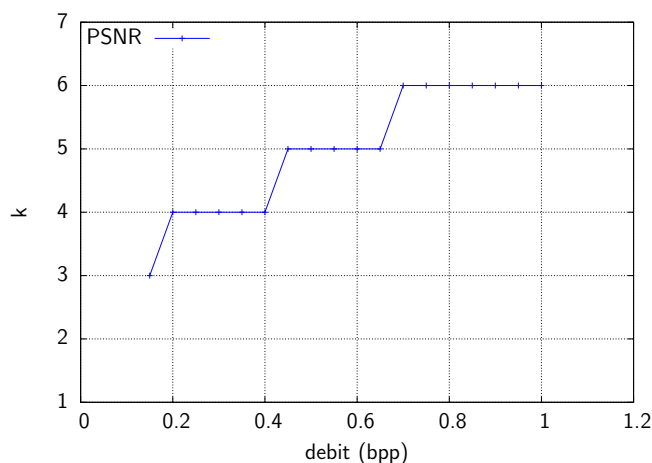


FIGURE 3.11: Valeur moyenne de  $k$  en fonction de débit pour la métrique PSNR

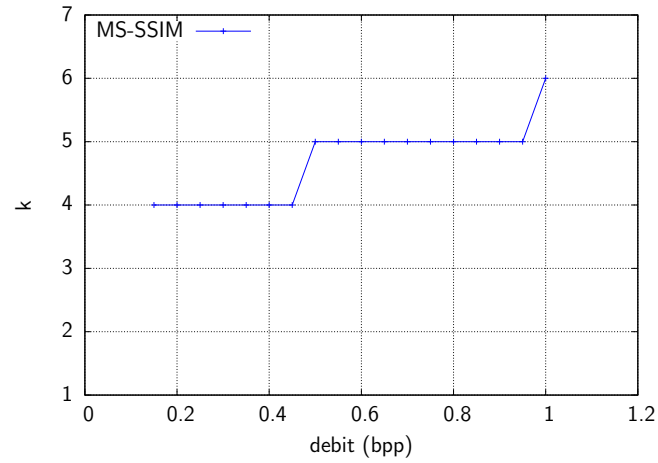
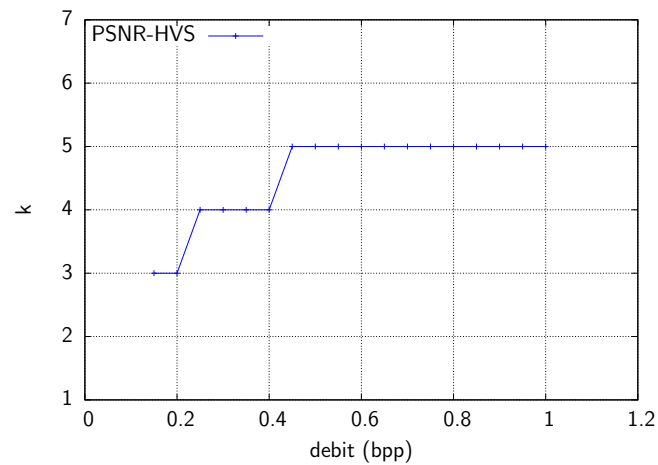
Dans un premier essai, nous avons utilisé le *PSNR* pour mesurer la qualité de l'image. La FIGURE 3.11 représente les résultats obtenus. Comme une interprétation générale, nous recommandons de choisir  $k = 6$ , pour des débits binaires entre 0,7 et 1,0 bpp,  $k = 5$  entre 0,45 et 0,7 bpp,  $k = 4$  entre 0,2 et 0,45 et  $k = 3$  pour un débit au dessous de 0,2 bpp.

Dans une deuxième expérience, nous avons choisit les métriques qui se basent sur le système visuel humain : *PSNR-HVS* et *M-SSIM*. En moyennant les résultats, nous avons pu déterminé la valeur optimale statistiques pour un débit donné comme on a fait lors de notre première expérience avec la métrique PSNR.

Les résultats obtenus sont présentés dans la FIGURE 3.12 et FIGURE 3.13. Comme on peut l'interpréter, dans le cas de *M-SSIM* nous recommandons de choisir  $k = 6$  pour un débit entre 0,95 et 1 bpp,  $k = 5$  pour un débit entre 0,45 et 0,95,  $k = 4$  pour un débit au dessous de 0,45 bpp.

Dans le cas de PSNR-HVS nous recommandons de choisir  $k=5$  pour un débit entre 0,4 et 1 bpp,  $k = 4$  pour un débit entre 0,4 et 0,2,  $k = 3$  pour un débit en dessous de 0,2 bpp.

Puisqu'il est difficile de concevoir une métrique capable de faire face à plusieurs types de distorsions, nous avons moyenné les  $k$  moyen de PSNR, M-SSIM et PSNR-HVS afin

FIGURE 3.12: Valeur moyenne de  $k$  en fonction de débit pour la métrique M-SSIMFIGURE 3.13: Valeur moyenne de  $k$  en fonction de débit pour la métrique PSNR-HVS

d'avoir plus de confiance dans notre jugement sur le choix de la meilleure valeur statistique de  $k$  en fonction du débit.

La courbe donnant la valeur moyenne de  $k$  en fonction du débit binaire est présentée dans la FIGURE 3.14.

En résumé, nous recommandons de choisir :

- $k = 6$  pour un débit entre 0,95 et 1 bpp
- $k = 5$  pour un débit entre 0,45 et 0,95 bpp
- $k = 4$  pour un débit entre 0,2 et 0,45 bpp
- $k = 3$  pour un débit en dessous de 0,2 bpp

Afin d'évaluer visuellement la qualité des images avec les bonnes valeurs de  $k$  choisies pour les débits entre 0,45 et 1 bpp, nous avons comparé la qualité de différentes images. Comme montré, dans la FIGURE 3.16, qu'il n'y a pas une dégradation dans la qualité visuelle des images avec  $k = 6$  et  $k = 5$ .

Cependant, à bas débit (0,2 bpp), l'image est segmentée en bloc grossiers, pauvres en informations pertinentes ( voir FIGURE 3.15 pour JPEG,  $k = 4$  et  $k = 3$ ). Nous pouvons noter qu'à bas débit, les valeurs des métriques de mesure de qualité utilisées restent inchangées.

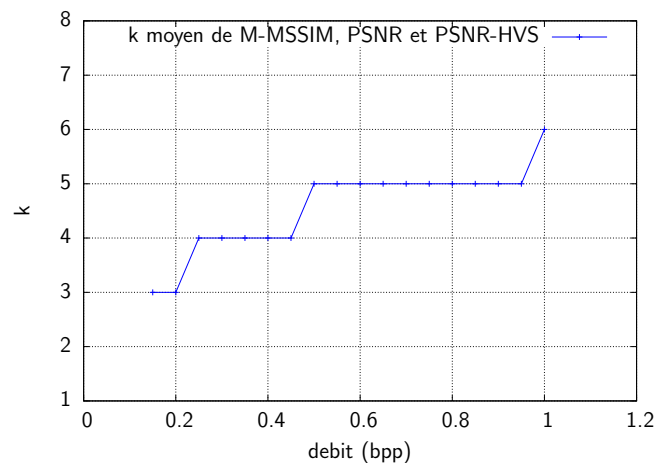


FIGURE 3.14:  $k$  moyen de PSNR, PSNR-HVS et MSSIM pour différentes valeurs de débit d'une base de 60 images

## 3.6 Conclusion

Dans ce chapitre nous avons étudié l'impact de la diminution du paramètre  $k$  sur la qualité visuelle de l'image. Nous nous sommes basés sur le fait que la qualité visuelle de



(a) Lena (JPEG),  $PSNR = 27,33$  dB,  $PSNR - HVS = 23,9$  dB,  $MS - SSIM = 0,74$   
 (b) Lena ( $k = 4$ ),  $PSNR = 27,32$  dB,  $PSNR - HVS = 23,89$  dB,  $MS - SSIM = 0,74$   
 (c) Lena ( $k = 3$ ),  $PSNR = 27,19$  dB,  $PSNR - HVS = 23,77$  dB,  $MS - SSIM = 0,73$



(d) Plane (JPEG),  $PSNR = 25,18$  dB,  $PSNR - HVS = 23,89$  dB,  $MS - SSIM = 0,74$   
 (e) Plane ( $k = 4$ ),  $PSNR = 25,1$  dB,  $PSNR - HVS = 21,67$  dB,  $MS - SSIM = 0,74$   
 (f) Plane ( $k = 3$ ),  $PSNR = 25,06$  dB,  $PSNR - HVS = 21,41$  dB,  $MS - SSIM = 0,74$

FIGURE 3.15: Image "Lena" ( $512 \times 512$  pixels) et "Plane" ( $768 \times 512$  pixels) codées à  $0,2$  *bpp*



(a) Lena  $512 \times 512$  ( $k = 5$ ),  $PSNR = 35,82$  dB,  $PSNR - HVS = 43,39$  dB,  $MS - SSIM = 0,92$



(b) Lena  $512 \times 512$  ( $k = 6$ ),  $PSNR = 36,41$  dB,  $PSNR - HVS = 43,42$  dB,  $MS - SSIM = 0,93$



(c) Peppers  $512 \times 512$  ( $k = 5$ ),  $PSNR = 33,92$  dB,  $PSNR - HVS = 41,98$  dB,  $MS_SSIM = 0,89$



(d) Peppers  $512 \times 512$  ( $k = 6$ ),  $PSNR = 34,78$  dB,  $PSNR - HVS = 42,27$  dB,  $MS - SSIM = 0,89$



(e) Plane  $768 \times 512$  ( $k = 5$ ),  $PSNR = 33,19$  dB,  $PSNR - HVS = 42,17$  dB,  $MS - SSIM = 0,93$



(f) Plane  $768 \times 512$  ( $k = 6$ ),  $PSNR = 35,01$  dB,  $PSNR - HVS = 44,63$  dB,  $MS - SSIM = 0,94$

FIGURE 3.16: Image Lena, Peppers et Plane codées à  $0,95$  bpp



FIGURE 3.17: Images d'évaluation



l'image est fonction du débit binaire de compression, car plus l'image est compressée, plus les composantes dans un bloc deviennent non significatifs (égaux à zéro). Elle dépend aussi des caractéristiques de l'image (luminance, la texture, etc.) et de la façon avec laquelle cette qualité est évaluée. Nous avons testé la chaîne de compression que nous avons proposée dans le chapitre 2 et trois métriques pour l'évaluation de la qualité des images ont été utilisées : le rapport signal sur bruit (*PSNR*), l'indice multi-échelle de qualité (*MS-SSIM*) et *PSNR-HVS* sur une base de donnée de 60 images. Le choix de  $k$  en fonction de débit pour chacune de ces métriques de mesure de qualité a été présenté. Enfin, en moyennant les valeurs de  $k$ , nous avons proposé un  $k$  optimal statistique pour chaque débit.

Nous avons constaté que le paramètre  $k$  peut être diminué selon le débit sans perdre significativement de la qualité de l'image. À  $0,2$  *bpp* par exemple, *JPEG* donne un *PSNR* de  $27,33$  *dB*,  $PSNR - HVS = 23,9$  *dB* et un *MSSIM* de  $0,74$  pour l'image Lena. Avec notre méthode, pour un débit égale à  $0,2$  *bpp*, avec le paramètre  $k = 4$ , nous obtenons une qualité visuelle d'image proche de celle obtenue avec *JPEG* ( $PSNR = 27,33$  *dB*,  $PSNR - HVS = 23,89$  *dB* et un  $M - SSIM = 0,74$ ). Avec l'utilisation de notre méthode, le gain en coût de traitement observé est approximativement 85%.

Nous prenons, dans le chapitre suivant, les bonnes valeur de  $k$  pour une implantation matérielle afin de comparer les résultats avec ceux obtenus par une implantation logicielle.



# Chapitre 4

## Évaluation des performances

### Sommaire

---

<b>4.1</b>	<b>Introduction</b>	<b>105</b>
<b>4.2</b>	<b>Evaluation par analyse numérique</b>	<b>106</b>
<b>4.3</b>	<b>Évaluation par expérimentation sur un nœud réel</b>	<b>109</b>
4.3.1	Présentation de la plateforme utilisée	109
4.3.2	Banc d'essai utilisé pour les mesures	111
4.3.3	Estimation durée de vie d'un nœud capteur	112
<b>4.4</b>	<b>Implantation de la chaîne de compression sur circuit ASIC</b>	<b>115</b>
<b>4.5</b>	<b>Conclusion</b>	<b>118</b>

---

### 4.1 Introduction

Nous avons étudié dans le chapitre 2 et 3 des solutions de compression d'images basées sur la DCT qui réduisent le temps d'exécution et donc la quantité d'énergie consommée par le microcontrôleur. Nous avons en effet prouvé que l'utilisation de la DCT de Cordic-Loeffler (méthode de transformation sans multiplications) combinée avec une approche zonale contribue fortement à réduire le nombre de cycles nécessaires au microcontrôleur pour compresser les données de l'image.

Ce chapitre a pour objectif d'évaluer les performances de nos propositions, en particulier d'évaluer le coût en énergie sur un cycle de fonctionnement <capture - compression-transmission> et estimer le gain en termes de durée de vie du capteur d'images. Cette évaluation a été réalisée dans un premier temps par analyse numérique. Les résultats obtenus donnent une estimation basse de la consommation d'énergie car l'analyse numérique ne prend en compte que l'activité du microcontrôleur et de l'émetteur radio. Tous les autres composants (y compris la mémoire) sont omis. L'évaluation est donc simpliste

mais cela permet de montrer que les caractéristiques du microcontrôleur et de l'émetteur radio vont avoir un impact sur le rentabilité énergétique de la compression d'image au niveau du capteur sans fil. Nous avons ensuite mesuré la consommation d'énergie et le temps d'exécution sur un capteur réel composé d'un nœud Mica2 et d'une caméra Cyclops. Ces mesures permettent d'évaluer le coût énergétique en prenant en compte tous les composants électroniques intervenant dans le capteur sans fil.

Dans ce chapitre, nous étudions les résultats obtenus. Enfin, nous présentons les performances de notre chaîne de compression d'image lorsque celle-ci est implantée sur circuit en considérant la technologie *ASIC*.

## 4.2 Evaluation par analyse numérique

L'analyse numérique est une technique simple et rapide d'évaluer le temps d'exécution et la consommation d'énergie des composants électroniques sans aucun matériel, en exploitant directement les informations relatives à la puissance consommée fournies dans les fiches constructeurs. Cette approche pour l'évaluation de performance a été utilisée dans (Lee *et al.*, 2007) par exemple. Quelques calculs suffisent pour les besoins de l'évaluation. Ils considéraient deux composants électroniques seulement, le microcontrôleur et l'émetteur radio.

Au niveau du microcontrôleur, le temps d'exécution et le coût d'énergie consommée par une tâche sont fonction du nombre de cycles à exécuter ( $C$ ), de la puissance consommée ( $P$ ) par le circuit et de la fréquence de son horloge ( $F$ ). Le temps d'exécution du microcontrôleur, noté  $t_{MCU}$ , est donné par la formule :

$$t_{MCU} = \frac{C}{F} \quad (4.1)$$

avec  $t_{MCU}$  en  $\mu s$  et  $F$  en  $MHz$ .

L'énergie consommée par le microcontrôleur, notée  $e_{MCU}$ , est le produit de la puissance consommée et du temps d'exécution :

$$e_{MCU} = t_{MCU} \times P = \frac{C \times P}{F} \quad (4.2)$$

avec  $e_{MCU}$  en  $\mu J$ ,  $P$  en  $mW$  et  $t_{MCU}$  en  $\mu s$ .

Au niveau de l'émetteur radio, en considérant le canal de communication dans sa plus simple expression, le temps de transmission et le coût d'énergie sont fonction du nombre de bits à transmettre ( $V$ ), de la puissance consommée ( $P$ ) par l'émetteur radio et débit

de l'émetteur ( $D$ ). Le temps de transmission est donné par l'équation :

$$t_{Tx} = \frac{V}{D} \quad (4.3)$$

avec  $t_{Tx}$  exprimé en  $s$ , le débit ( $D$ ) en bits par seconde (bps) et  $V$  en bits.

L'énergie consommée par l'émetteur radio pour la transmission de  $V$  bits, notée  $e_{Tx}$ , est le produit de la puissance et du temps de transmission :

$$e_{Tx} = t_{Tx} \times P = \frac{V \times P}{D} \quad (4.4)$$

avec  $e_{MCU}$  en  $mJ$ ,  $P$  en  $mW$  et  $t_{Tx}$  en  $s$ .

Puisque ce coût d'énergie dépend fortement des caractéristiques de la plateforme utilisée, nous avons pris comme référence les valeurs caractéristiques de nœuds-capteurs bien connus : Mica2 et MicaZ fabriqués par Crossbow. Ils s'appuient sur le même microcontrôleur, un *Atmel ATMEGA 128L* cadencé à  $7,3728 MHz$ , mais les émetteurs radio sont différents. Mica2 utilise un circuit ChipCon CC1000 qui opère à la fréquence de  $915 MHz$  et fournit un débit de transmission de  $38,4 kbps$ . MicaZ utilise un ChipCon CC2420 conforme à la norme IEEE 8.2.15.4 qui opère à  $2,4 GHz$  pour un débit de transmission de  $250 kbps$ .

Les valeurs numériques que nous avons adoptées pour caractériser les composants sont tirées des travaux de (Polastre *et al.*, 2005). Ces valeurs sont données dans la TABLE 4.1 (pour un rapport de puissance de l'émetteur radio de  $0 dBm$ ).

Mote	Mica2	MicaZ
Processeur	ATmega128	ATmega128
Radio	CC1000	CC2420
Fréquence d'horloge [MHz]	7,3728	7,3728
Débit [kbps]	38,4	250
Processeur [mW]	22	22
Transmission [mW]	69	57
Réception [mW]	41	63

TABLE 4.1: Spécifications des nœuds Mica2 et MicaZ

Notre analyse numérique considère une image monochrome de  $128 \times 128$  pixels codée à l'origine sur 8 bits par pixel. Elle a donc une taille de 16384 octets quand elle n'est pas compressée. Les résultats de l'analyse numérique sont donnés dans le TABLE 4.2.

Nous pouvons remarquer que pour le nœud Mica2, la compression est souhaitable du point de vue de la durée de temps et de l'énergie. Les économies d'énergie et de temps deviennent de plus en plus intéressants lorsque l'on augmente le taux de compression.

Les résultats pour le nœud MicaZ sont plus nuancés. Pour les images peu compressées (entre 0,5 *bpp* et 1 *bpp*), la compression n'est pas rentable pour le nœud source puisque l'image non compressée consomme moins d'énergie. Comme le montre dans le TABLE 4.2. A 0,7 *bpp* par exemple, la compression toute seule (sans transmission) consomme (34,33 *mJ*) qui représente plus d'énergie que la transmission sans compression (29,88 *mJ*).

	Cycles	Compression		Données [bits]	Transmission Mica2		Transmission MicaZ	
		Temps [s]	Énergie [mJ]		Temps [s]	Énergie [mJ]	Temps [s]	Énergie [mJ]
Sans compression	-	-	-	131072	3,41	235,52	0,52	29,88
1 <i>bpp</i> (k=6)	13361920	1,81	39,87	16384	0,43	29,44	0,07	3,73
0,7 <i>bpp</i> (k=5)	11502592	1,56	34,33	11469	0,3	20,60	0,05	2,61
0,5 <i>bpp</i> (k=5)	11502592	1,56	34,33	8192	0,21	14,72	0,03	1,87
0,2 <i>bpp</i> (k=3)	8758784	1,19	26,14	3276	0,08	5,89	0,013	0,75

TABLE 4.2: Le coût en énergie et en temps de la compression et de la transmission d'une image monochrome  $128 \times 128$

La FIGURE 4.1 illustre le coût en énergie de la compression et de la transmission pour les deux plateformes. Les parties haute et basse de chaque barre de la figure représentent respectivement le coût d'énergie de la transmission et du traitement de l'image. L'énergie consommé pour la transmission d'une image  $128 \times 128$  sur un nœud Mica2 vaut 235,52 *mJ*. Nous présentons les résultats pour différents débits de compression ainsi que pour le cas de la transmission de l'image sans compression. Nous observons que compresser avant la transmission n'est pas toujours rentable en termes de consommation d'énergie, cela dépend de la combinaison microcontrôleur/ émetteur radio qui est utilisée.

Dans le cas d'un nœud Mica2, en combinaison avec le microcontrôleur ATMEGA128 et l'émetteur radio CC1000, les résultats montrent qu'il est toujours plus avantageux de compresser l'image avant transmission. Les économies d'énergie sont d'environ 88%, 91%, 94% et 97% respectivement, quand on transmet une image codée au débit binaire de 1 *bpp*, 0,7 *bpp*, 0,5 *bpp* et 0,2 *bpp*.

Mais les résultats avec le nœud MicaZ montrent un résultat contraire, c'est à dire qu'il est plus rentable de transmettre l'image sans compression pour conserver l'énergie. La transmission d'une image monochrome de taille  $128 \times 128$  compressée à un débit binaire de 1 *bpp* présente un excès d'énergie par rapport à la transmission de l'image sans compression d'environ 46%. Cet excès est lié au composant radio CC2420 qui fournit un débit de transmission nettement plus élevé que le CC1000 utilisé par le nœud Mica2.

Pour conclure, les résultats montrent sans ambiguïté que la compression n'est pas toujours rentable pour le capteur source. Toutefois, la compression sera toujours profitable pour les nœuds relayant les paquets jusqu'au puits. Elle peut donc être un moyen de

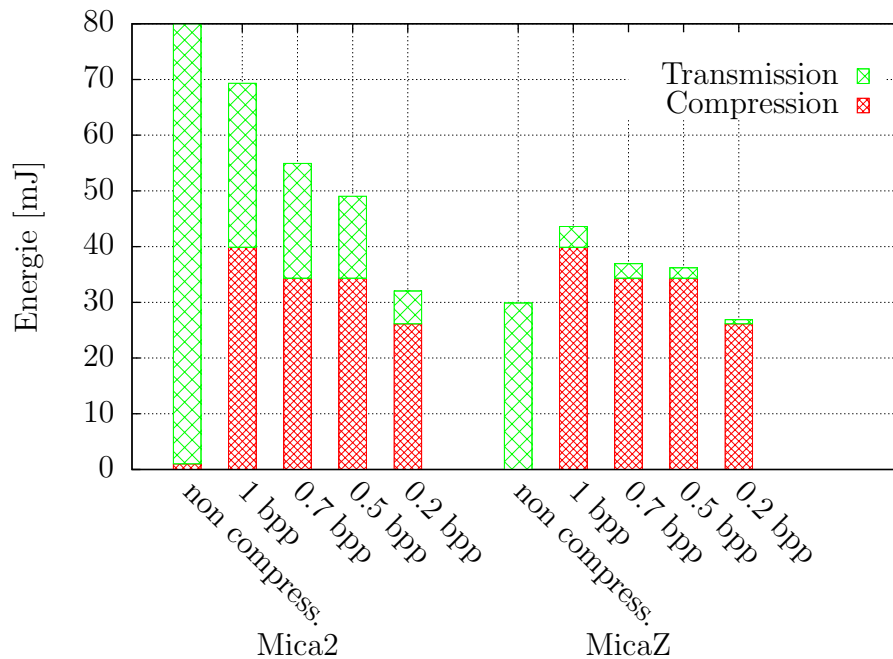


FIGURE 4.1: Énergie de compression et de transmission d’une image  $128 \times 128$  pour des débits 0.2, 0.5, 0.7 et 1 bpp

ralentir le phénomène du trou d’énergie à proximité du puits.

L’approche par analyse numérique ne présente pas réellement la consommation d’énergie puisque les mesures ne prennent pas en considération par exemple la consommation d’énergie lors de la lecture et l’écriture des données en mémoire. Pour cette raison, nous avons implanté notre l’algorithme sur un vrai capteur d’image afin de mesurer la consommation réelle.

## 4.3 Évaluation par expérimentation sur un nœud réel

### 4.3.1 Présentation de la plateforme utilisée

Le nœud Mica2 (Crossbow, n.d.) montré sur la FIGURE 4.2 dispose d’une unité de traitement composée d’un microcontrôleur (MCU) ATmega128L de Atmel avec 512 *Ko* de mémoire flash pour les données, d’une EEPROM de 4 *Ko* et d’une unité de communication basée sur l’émetteur radio Chipcon CC1000. Un connecteur 51 broches permet de connecter des cartes filles supportant les capteurs proprement dits. Le nœud est alimenté en 3V avec deux piles AA.

La caméra Cyclops a été conçue par le constructeur Agilent et le CENS (Center for Embedded Network Sensing) de l'UCLA sous la forme d'une carte fille qui peut être enfichée sur un nœud Mica2 ou MicaZ. Elle prend en charge la capture et le traitement d'images monochrome à 8 *bpp* ou couleur en format RGB à 24 *bpp* ou au format YCbCr à 16 *bpp* avec une relativement faible consommation d'énergie. Cyclops est composé d'une caméra CMOS de moyenne qualité ADCM-1700 (Agilent), un microcontrôleur (MCU) ATmega128L de Atmel avec 128 *Ko* de mémoire flash, 4 *Ko* de mémoire RAM et un circuit logique programmable Xilinx XC2C256. Une caméra Cyclops attachée à un mote Mica2 est visualisée FIGURE 4.3.



FIGURE 4.2: Un nœud capteur Mica2

Les motes Mica2 fonctionnent sous TinyOS/nesC développé par l'UC-Berkeley (UC Berkeley, n.d.). La taille maximale des messages imposée par TinyOS est de 255 octets (Thorn, 2005).



FIGURE 4.3: Une caméra Cyclops attachée à un mote Mica2

Les applications de compression et de transmission d'image ont été développées en NesC sous TinyOS (Levis *et al.*, 2005).

Les programmes que nous avons développés étaient divisés en deux parties :

- La partie capture et traitement (avec/ sans compression, mise en forme des paquets des données) était téléchargée dans la carte fille Cyclops.



- La partie communication (transmission des paquets) était téléchargée sur le nœud Mica2.

Le transfert des paquets formés par le Cyclops au nœud Mica2 s’effectuait par le bus I2C. Bien que TinyOS peut traiter des paquets jusqu’à 255 octets de long, nous avons conservé la taille maximum recommandée par défaut qui est de 36 octets (compromis énergie/ taux d’erreur). Sur ces 36 octets, 7 sont réservés par le protocole MAC, le champs de données ne peut donc dépasser 29 octets. Le champs de données est constitué d’un entête de 5 octets (2 octets pour le numéro d’image, 1 octet pour le type de codage et 2 octets pour l’offset du premier élément de donnée), et donc les données proprement dite (un nombre entier de pixels ou de bloc  $8 \times 8$  compressés selon le scénario) ne peuvent dépasser 24 octets.

### 4.3.2 Banc d’essai utilisé pour les mesures

Nous utilisons le banc d’essai schématisé FIGURE 4.4 pour mesurer les performances de notre application en termes de temps d’exécution et de consommation d’énergie.

Ce banc d’essai permet de mesurer la puissance consommée par le nœud capteur en fonction du temps sur la durée d’un cycle de fonctionnement comprenant la capture d’une image, le traitement des données, la paquetisation des données et la transmission des paquets. Dans le banc d’essai, une résistance de petite valeur ( $R_I = 1\Omega$ ) est connectée en série avec le dispositif sous évaluation (DSE, c’est-à-dire, notre capteur d’image composé d’un mote Mica2 et une caméra Cyclops). Une alimentation à tension continue remplace les piles. Les tensions sur l’alimentation ( $V_1 \approx 3 \text{ Volts}$ ) et sur la résistance ( $V_2$ ) sont récupérées toutes les  $0.5ms$  en utilisant un oscilloscope numérique *Agilent 54622A*.

Des calculs simples donnent le courant qui circule à travers le DSE et, à partir de cela, la puissance consommée instantanée ( $P$  en Watts). A l’instant  $t$ ,  $P(t)$  est donné par :

$$P(t) = \frac{[V_1(t) - V_2(t)] \cdot V_2(t)}{R_I} \quad (4.5)$$

Donc, la quantité d’énergie consommée par le DSE ( $E$  en Joules) entre les instants  $T_A$  et  $T_B$  est calculée par :

$$E = \int_{T_A}^{T_B} P(t) \cdot dt \quad (4.6)$$

Un exemple de trace enregistrée est montré FIGURE 4.5. Cette trace se rapporte à un cycle de fonctionnement du nœud capteur incluant la capture d’une image monochrome de  $64 \times 64$  pixels, la compression à un débit de 0.5 bpp et la transmission des paquets

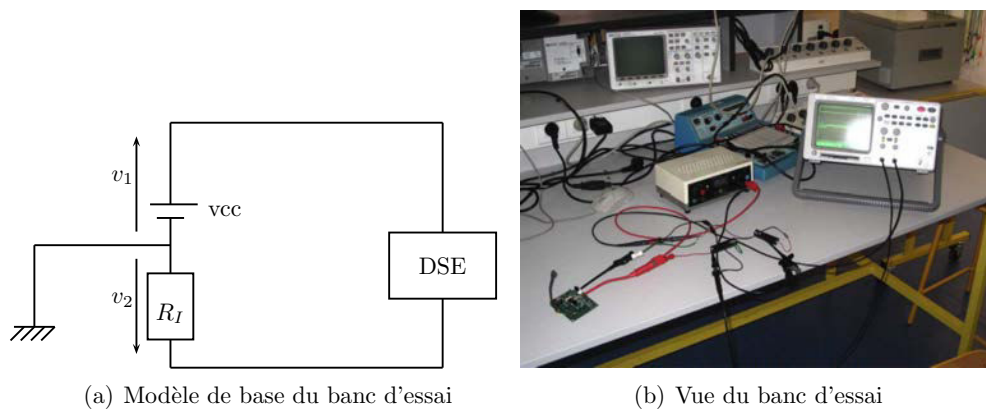


FIGURE 4.4: Banc d'essai utilisé pour les mesures.

avec un rapport puissance de transmission de  $0 \text{ dBm}$ . Le signal observé est représentatif de toutes les traces obtenues pendant nos expérimentations. Dans une première phase, un augmentation rapide de la puissance consommée signale l'étape de capture de l'image grâce à l'activation d'une LED. Cette LED est éteinte immédiatement après la capture de l'image pour ne pas biaiser la mesure de la puissance consommée pendant la compression et la transmission de l'image. Nous observons que la capture de l'image (avec un Mica2 connecté) est gourmande en termes de puissance consommée, en partie due à l'activation de la LED. La caméra consomme une puissance d'environ  $60.17 \text{ mW}$  pendant la capture, sur une durée approximativement de  $0.97$  secondes, soit un coût d'énergie d'environ  $59 \text{ mJ}$ . Nous avons noté que l'énergie consommée pour la capture d'image ne varie pas de manière significative d'une image à l'autre.

Après la capture de l'image, nous pouvons reconnaître l'étape de traitement et de transmission des données par paquets. Dans la trace présentée FIGURE 4.5, nous pouvons clairement identifier la transmission de 13 paquets obtenus comme le résultat de la compression à  $0.5 \text{ bpp}$  (pics associés au basculement en mode émission du composant radio). Le temps d'exécution utilisé pour le traitement et la transmission des données est d'environ  $1.18$  secondes, et la consommation d'énergie est d'environ  $106.5 \text{ mJ}$ . Le cycle d'exécution complet pour le scénario pris en exemple dure approximativement  $2.15$  secondes et consomme  $165.5 \text{ mJ}$ .

### 4.3.3 Estimation durée de vie d'un nœud capteur

Dans cette partie, nous appliquons les modèles de consommation d'énergie pour évaluer la durée de vie du nœud capteur d'image pour la transmission et la compression d'une image monochrome avec la chaîne de compression que nous avons proposé et comparer ce

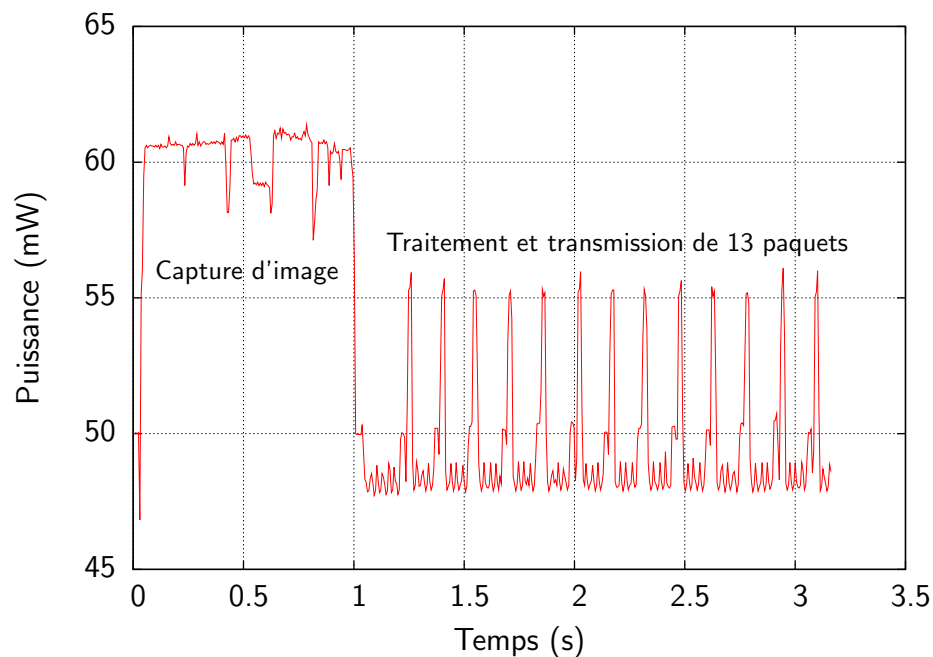


FIGURE 4.5: Trace de la puissance consommée par le capteur pour une image monochrome  $64 \times 64$  compressée à 0,5 bpp

coût avec celui d'une chaîne de compression basée sur JPEG.

Un premier scénario consiste à transmettre une image monochrome de taille  $128 \times 128$  sans compression. Dans ce scénario, les données occupent 16384 octets, la source doit donc préparer 682 paquets de 36 octets chacun et un paquet de 28 octets.

Un deuxième scénario consiste à appliquer la compression JPEG classique à un débit binaire de 0,5 bpp et transmettre l'image compressée. Dans ce cas la source va préparer 52 paquets de taille variable puisqu'ils doivent contenir toujours un nombre entier de blocs compressés.

Nous avons aussi considéré le scénario qui consiste toujours à compresser l'image avant la transmission, mais en adoptant cette fois-ci l'approche zonale ( $k=5$ ) pour réduire le coût de la compression. L'image à transmettre est aussi codée à 0,5 *bpp*.

Les performances de tous ces scénarios sont étudiées en prenant un scénario en exemple où le capteur d'image va capturer et transmettre au puits une image monochrome  $128 \times 128$  toutes les heures. Ils va donc être activé une fois par heure, exécuter un cycle de travail, et retourner à l'état endormi jusqu'au prochain réveil pour préserver sa batterie. La consommation d'énergie du capteur en mode sommeil est de  $75 \mu W$ . Le capteur est alimenté par 2 piles AA d'une capacité de 2000 *mAh*.

L'énergie consommée pour la transmission de l'image non comprimée est égale à

2452  $mJ$ . Cette valeur va nous servir de référence pour calculer les gains en durée de vie que peuvent être obtenus avec la chaîne de compression que nous proposons.

	Temps (ms)	Énergie (mJ)	Durée de vie (jours)
Sans compression	33561	2452	331
CL (ref.)	8010	525	1133
CL.Z ( $k=5$ )	6327	417	1311

TABLE 4.3: Temps d'exécution et énergie consommée de la compression et la transmission pour une image  $128 \times 128$  codée à 0.5 bpp sur le nœud caméra sans fil (la durée de vie du nœud donnée est dans le cas où le nœud envoie une image par heure)

Avec le scénario de transmission des images sans compression la durée de vie des capteurs d'image serait de 331 jours équivalent à la transmission de 7944 images. Avec l'application de notre chaîne de compression au débit de 0,5  $bpp$  et en choisissant  $k = 5$ , la durée de vie des capteurs serait de 1311 jours, et 31464 images auraient été transmises. Grâce à notre chaîne de compression, le capteur va donc pouvoir fonctionner quasiment 4 fois plus longtemps.

Les résultats obtenus TABLE 4.3 montrent que le gain est d'environ **980 jours** en utilisant l'algorithme de la compression basée sur CL.Z (Cordic Loeffler Zonale) (avec  $k = 5$ ) par rapport à la transmission d'image sans compression. Comparé à l'algorithme de compression JPEG standard (sans approche zonale), notre proposition permet de prolonger la durée de vie du capteur de 178 jours, soit environ 6 mois. C'est loin d'être négligeable.

Les résultats confirment que la compression de l'image amène des économies d'énergie importantes avec la plateforme considérée. Le cas avec notre chaîne de compression implique plus d'économie d'énergie que la compression JPEG classique.

On constate qu'il y a une grande différence de résultats en comparant les valeurs obtenues par une analyse numérique et une expérimentation sur un capteur réel. À titre d'exemple, le résultat analytique de la consommation d'énergie pour ( $k = 5$ ) d'une image  $128 \times 128$  était égale à 49.02  $mJ$ , ce qui représente une valeur **8,5 fois** plus petite que celle obtenues lors des mesures sur un nœud capteur réel.

Tous ces résultats correspondent à une implantation logicielle de notre chaîne de compression. Pour économiser encore plus d'énergie et de temps de traitement, il faudrait implanter l'algorithme de compression sur un circuit plutôt que de la faire exécuter par le microcontrôleur.

## 4.4 Implantation de la chaîne de compression sur circuit ASIC

La réduction de la complexité obtenue avec la chaîne de compression proposée a entraîné une amélioration significative des performances (en temps d'exécution et en consommation d'énergie) dans son implantation logicielle sur un nœud capteur composé d'un Mica2 et d'une caméra Cyclops. Il reste à étudier si l'approche zonale a aussi un impact significatif sur les performances dans une implantation sur circuit.

Cette partie a fait l'objet d'une coopération Franco-Tunisienne de Janvier 2009 à décembre 2011 (projet CNRS-DGRST 09/R 11-22, n°22670, intitulé "Compression rapide et robuste d'images dans les réseaux de capteurs sans fil") entre le CRAN et le laboratoire d'électronique et de Micro-électronique de Monastir. L'objectif scientifique était de définir, concevoir, et au final implanter sur circuit intégré ASIC (technologie CMOS submicronique de 45 nm) un algorithme de compression d'images satisfaisant les contraintes inhérentes aux réseaux de capteurs sans fil, c'est à dire qui ait une très faible complexité (contraintes de conception du circuit et consommation d'énergie) et une très forte tolérance aux pertes de paquets par le support de transmission. Notre chaîne de compression basée sur une approche zonale fournissait une réponse au problème de la réduction de la complexité. L'augmentation de la tolérance aux pertes de paquets, qui est un problème en dehors du périmètre du présent mémoire (nous ne donnerons donc pas de détails ici), était obtenue en traitant les blocs de l'image dans un ordre chaotique pour que les paquets de données à transmettre ne contiennent pas de blocs contigus (voir (Duran-Faundez et Lecuire, 2008) et (Duran-Faundez *et al.*, 2011) pour plus d'informations). Le CRAN ne possède pas de savoir-faire dans la conception sur circuit et la complémentarité avec l'équipe tunisienne se situe principalement à ce niveau.

Le circuit qui a été conçu intègre une mémoire de type SRAM qui stocke les données de l'image capturée par le module caméra. Cette mémoire peut contenir des images monochromes encodées sur 8 *bpp* de taille 128 × 128 maximum. Le circuit délivre en sortie vers le microcontrôleur des paquets préformatés contenant un ou plusieurs blocs d'images compressées, de taille maximum de 29 octets, entête compris. Le format des paquets est le même que celui adopté dans l'implantation logicielle, et donc l'entête fait 5 octets (2 octets pour le numéro d'image, 1 octet pour le type de codage et 2 octets pour l'offset du 1<sup>er</sup> élément de donnée). L'architecture du circuit est schématisé FIGURE 4.6. Elle est composée de trois modules, un pour la DCT, un pour la quantification et un pour le codage et la mise en paquets des données. Ces trois modules se partagent une mémoire

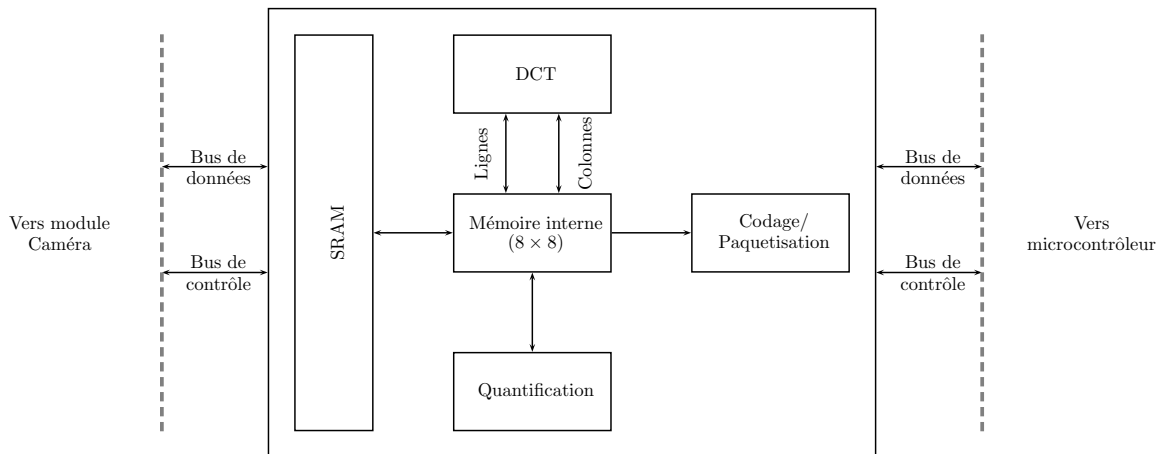


FIGURE 4.6: Architecture en bloc du circuit de compression

interne stockant les résultats intermédiaires de traitement d'un bloc d'image de  $8 \times 8$  éléments. Les détails de la conception du circuit sont donnés dans le mémoire de thèse de M. Kaddachi (Kaddachi, 2012).

Les différents blocs du circuit de compression ont été décrits par un ensemble de machines à états finis étendus (EFSM) au niveau transfert de registre (MTL) en utilisant le langage VHDL. La synthèse du circuit a été faite à l'aide de la suite d'outils Synopsys en utilisant sa bibliothèque de cellules standard. L'outil cadence a été utilisé pour les opérations de placement et d'interconnexions des cellules. La FIGURE 4.7 donne le dessin de masque du circuit de compression proposé avec la technologie  $45 \text{ nm}$ . La mémoire SRAM adoptée (ST-SPHD-16384x16m32-LTB1) est proposée par la société ST-Microelectronics. Cette mémoire a été choisie pour son temps d'accès rapide et à sa faible consommation.

Les critères de performance des circuits sont principalement liés à sa surface (plus on arrive à réduire la complexité du circuit et moins on utilise de cellules standard, et par conséquent plus la surface du circuit peut être petite), la puissance consommée et la fréquence de fonctionnement (plus la fréquence de fonctionnement est élevée, et plus le circuit travaille vite).

La TABLE 4.4 fournit les performances du circuit de compression pour des conceptions considérant  $k = 6$  (pour des débits de compression entre 0,95 et 1bpp),  $k = 5$  (débit de compression entre 0,45 et 0,95 bpp) et  $k = 4$  (débit inférieur à 0,45 bpp).

Les performances pour une conception classique (sans approche zonale, c'est-à-dire  $k = 8$ ) de notre chaîne de compression sont aussi données pour comparaison.

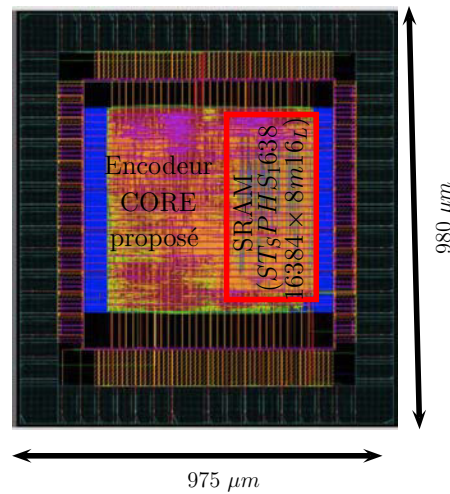


FIGURE 4.7: Dessin de masque du circuit type JPEG proposé

	$F_{max}(MHz)$	Surface occupée ( $mm^2$ )	Puissance dynamique (mW)	Puissance statique ( $\mu W$ )
Encodeur classique	266	0,23	13,35	222,56
Encodeur proposé (k=6)	333	0,217	16,11	226
Encodeur proposé (k=5)	337,66	0,214	17,88	236,4
Encodeur proposé (k=4)	360	0,209	18,02	242,52

TABLE 4.4: Caractéristiques du circuit de compression pour la technologie ASIC 45 nm

Le codeur proposé pour le cas où  $k = 4$ , est plus performant que  $k = 6$  et  $k = 5$ . Il présente la fréquence de fonctionnement la plus élevée avec moins de surface occupée. C'est grâce à l'utilisation de l'algorithme Cordic-Loeffler qui est basé sur des opérations arithmétique sur des entier. De plus l'approche du codage en zonal contribue fortement à des meilleures économies dans ces grandeurs grâce à la diminution du nombre d'opérations.

Le circuit, une fois synthétisé avec 45 nm, atteint une période minimale de 2,77 ns, ce qui permet de traiter 453 M bits par seconde. Il occupe 0,209  $mm^2$  et consomme 18,02 mW. La fréquence la plus élevée est obtenue lorsque  $k$  est égal à 4. Avec cette, valeur, le circuit de compression atteint un débit binaire inférieur à 0,45 *bpp*.

Nous avons aussi évalué le coût de la consommation d'énergie et de temps de traitement d'image de taille  $128 \times 128$  pixels et compressée à 0,5 *bpp* avec le codeur que nous avons proposé. Nos résultats montrent que la consommation d'énergie varie entre 29,91 mJ pour le cas où  $k = 4$  et 34,31 mJ pour le cas où  $k = 6$ . L'utilisation du codeur proposé avec  $k = 4$  conduit à un gain de 12,8% par rapport au circuit dans le cas où  $k = 6$ . La même interprétation pourrait être considérée pour le temps d'exécution (le gain est d'environ 22% par rapport au cas où  $k = 6$ ).

Les meilleures performances du codeur proposé pour les réseaux de capteurs sans fil en termes de distorsion, de consommation d'énergie, de temps d'exécution et de coût en calcul sont clairement obtenues par l'approche du codage en zonal.

Technologie d'intégration ASIC	Solution	Surface occupée ( $K$ gates)	Fréquence (MHz)	Puissance dynamique
45 nm	(Kaddachi <i>et al.</i> , 2011)	368	266	50,1 $\mu W$ /MHz
	Notre proposition	256	360	50,05 $\mu W$ /MHz
0,6 $\mu m$	(Sun et Lee, 2003)	136,62	27	37,03 $\mu W$ /MHz
	Notre proposition	101,34	48,33	23,58 $\mu W$ /MHz
0,18 $\mu m$	(Meng-Chun <i>et al.</i> , 2006)	390	12,58	1,18 $\mu W$ /MHz
	(Zhou <i>et al.</i> , 2010)	–	16	2,5 $\mu W$ /MHz
	(Papadonikolakis <i>et al.</i> , 2007)	27,68	183	–
	Notre proposition	169,6	213	86,05

TABLE 4.5: Comparaison avec des solutions sur ASIC

Il existe dans la littérature différentes méthodes de compression d'image sur des circuits ASIC (Kaddachi *et al.*, 2011; Sun et Lee, 2003; Papadonikolakis *et al.*, 2007; Meng-Chun *et al.*, 2006; Zhou *et al.*, 2010). Ces solutions ont été implantées en utilisant les technologies 45 nm, 0,6  $\mu m$  et 0,18 nm. Les principales caractéristiques (surface, fréquence et la puissance) de ces circuits sont présentées dans le tableau 4.5 en les comparant avec le circuit que nous avons proposé.

On peut observer clairement que la méthode de compression que nous avons proposée, pour  $k = 4$  présente la fréquence de fonctionnement la plus élevée avec moins de surface occupée par rapport au  $k = 5$  et  $k = 6$ . Cela est grâce à l'utilisation de l'approche du codage en zonal combiné avec l'algorithme Cordic Loeffler.

## 4.5 Conclusion

Dans ce chapitre, nous avons évalué les performances de l'algorithme de compression proposé sur un nœud capteurs réel et nous avons pu obtenir un gain important en énergie. Nous avons ensuite implanté cet algorithme sur circuit intégré basé sur la technologie ASIC.

La chaîne de compression proposée offre des gains de performance significatifs, aussi bien dans une implantation logicielle (nœud programmable avec Nesc/Tinyos) que dans une implantation matérielle. Les résultats sont donc concluants.



# Conclusions et perspectives

Les réseaux de capteurs, nouvelle thématique de recherche innovante, viennent enrichir le domaine des réseaux et télécommunications. La dimension de ce nouveau type de réseau capable de collecter des informations sur un environnement et de les stocker ou les transmettre à une station collectrice, permet de surveiller différentes zones qui peuvent être hostiles à l'homme. La pluridisciplinarité de ce domaine pousse à la convergence des compétences de plusieurs laboratoires qui ont parfois des centres d'intérêts différents mais qui se rejoignent sur cette thématique. La particularité de ces nouveaux systèmes distribués émergents est la capacité réduite des noeuds en termes de calcul, de mémoire et d'énergie. Cette dernière caractéristique est le point critique dans ces réseaux. Les recherches dans ce domaine se focalisent donc dans la majorité des cas sur la gestion énergétique.

Pour optimiser la collecte et le transfert des images dans la zone de déploiement de réseau de capteurs sans fil, nous avons proposé une nouvelle méthode de compression d'image satisfaisant à l'une des contraintes induite par les réseaux de capteurs sans fil, c'est-à-dire de faible complexité. Cette compression d'image permet de réduire la taille des informations transmises tout en offrant une bonne qualité d'image et une faible consommation en énergie. Les critères de performance prenaient en compte le rapport débit-distorsion, le temps d'exécution ainsi que le coût d'énergie.

Notre algorithme de compression de données avec perte, s'inspirant du standard de compression JPEG introduit un nouveau paramètre  $k$  pour réduire le nombre de coefficients DCT par bloc de  $8 \times 8$  pixels à calculer, quantifier et encoder. Le choix de ce paramètre a fait l'objet d'une étude à l'aide de deux principaux critères : la consommation d'énergie sur le nœud caméra et les métriques de mesure de la qualité de l'image basée sur le système visuel humain.

L'étape de "décorrélation" de notre algorithme de compression combine la meilleure méthode de transformation par DCT (Cordic-Loeffler) et une approche zonale pour la sélection des coefficients. La méthode de transformation par DCT de Cordic-Loeffler vise

à réduire le nombre d'opérations par coefficient tandis que l'approche zonale permet de réduire le nombre de coefficients à traiter. Par conséquent, cette réduction du nombre de coefficients entraîne une économie d'énergie et du temps d'exécution sur toute la chaîne de compression puisque le nombre de coefficients à quantifier et coder est moins important.

L'algorithme de compression d'image proposé dans le cadre de cette thèse, comparé à l'algorithme de référence JPEG, a montré son efficacité en termes de consommation énergétique à taux de compression égal et à qualité d'image égale. De plus, cette proposition qui a été implantée sur un mote Mica2 équipé d'une caméra Cyclops, a démontré sa faisabilité pour des systèmes limités en ressources. Les performances obtenues sont probantes puisqu'on arrive à des économies très importante en termes d'énergie et du temps d'exécution.

L'implantation matérielle de la chaîne de compression sur circuit intégré a aussi été évaluée. Elle a été synthétisée sur des circuits basés sur la technologie ASIC et une comparaison avec d'autres solutions matérielles a été menée pour montrer l'efficacité de nos propositions.

Les travaux présentés tout au long de cette thèse ont traité de la compression et de la transmission d'images sur des réseaux de capteurs sans fil sous la contrainte de l'énergie. Plusieurs perspectives peuvent être envisagées sur la base de ces travaux.

Du point de vue du traitement des images, il faut s'intéresser à étudier les régions d'intérêt dans l'image parmi les méthodes populaires on peut citer les dérivées partielles d'ordre N (N-jets) ou les histogrammes locaux (histogrammes d'orientation des contours par exemple) et puis l'extraction de caractéristiques. Seulement la région d'intérêt sera compressée et transmise via le réseau. Il existe déjà plusieurs algorithmes de détecteurs de zone d'intérêt comme MSER (Maximally Stable Extremal Regions). D'autres algorithmes tels que SIFT comprennent à la fois une phase de détection et une phase d'extraction peuvent aussi être appliqués sur l'image à la source.

Du point de vue du réseau, des simulations des réseaux à grande échelle doivent être réalisées. Des études de différents protocoles de routage (mono et multichemins) et de couche MAC doivent être réalisées pour identifier quelles sont les meilleures stratégies pour les réseaux de capteurs d'image. Des expérimentations sur la palteforme de test Senslab<sup>1</sup>, sont aussi envisagées pour compléter les résultats de simulation.

Quelques renforcements sur la tolérance aux pertes de paquets peuvent aussi être étudiés par l'exploitation de mécanismes comme le tatouage d'images utilisé généralement

---

1. Le projet Senslab est un projet de recherche concernant les réseaux de capteurs à grandes échelles : <http://www.senslab.info/>

pour la protection des données informatiques (copies illégales, modification, etc.) (Chen *et al.*, 2003; Parisi *et al.*, 2004) mais qui peut aussi être utilisé pour embarquer dans les données de l'image compressée des informations redondantes utiles à sa reconstruction (Nikolaidis, 2012; Nyeem *et al.*, 2012).



# Liste des publications

## Revue internationale avec comité de lecture

- V. Lecuire, L. Makkaoui, J.-M. Moureau, *Fast zonal DCT for energy conservation in wireless image sensor networks*. Electronics Letters 48, 2 (2012) 125-127.
- L. Kaddachi, A. Soudani, V. Lecuire, K. Toriki, L. Makkaoui, J.M. Moureaux, *Low Power Hardware based Image Compression Solution for Wireless Camera Sensor Network*. Computer Standards & Interfaces 34, 1 (2012) 14-23.

## Conférences internationales avec comité de lecture

- L. Makkaoui, V. Lecuire et J.M. Moureaux, *Fast Zonal DCT-based image compression for Wireless Camera Sensor Networks*. International Conference on Image Processing Theory, Tools and Applications (IPTA10), page 126-129, France, Juillet 2010.
- L. Kaddachi, L. Makkaoui, A. Soudani, V. Lecuire, J.M. Moureaux, *FPGA-based image compression for low-power Wireless Camera Sensor Networks*. Third International Conference on Next Generation Networks and Services, 10-12 November 2011.

## Conférences nationales avec comité de lecture

- L. Makkaoui, V. Lecuire et J.M. Moureaux. , *Efficacité énergétique d'une DCT zonale rapide dans le contexte de la compression d'image dans les réseaux de capteurs sans fil*. CORESA, Compression et représentation des signaux audiovisuels, France, Septembre 2010.



# Bibliographie

- Ahmed, N., T. Natarajan et KR Rao (1974). Discrete cosine transform. *Computers, IEEE Transactions on*, **100**(1), 90–93.
- Ahn, G.S., S.G. Hong, E. Miluzzo, A.T. Campbell et F. Cuomo (2006). Funneling-mac : a localized, sink-oriented mac for boosting fidelity in sensor networks. In : *Proceedings of the 4th international conference on Embedded networked sensor systems*. ACM. pp. 293–306.
- Akkaya, K. et M. Younis (2005). A survey on routing protocols for wireless sensor networks. *Ad hoc networks*, **3**(3), 325–349.
- Akyildiz, I.F., T. Melodia et K.R. Chowdhury (2007). A survey on wireless multimedia sensor networks. *Computer networks*, **51**(4), 921–960.
- Akyildiz, I.F., W. Su, Y. Sankarasubramaniam et E. Cayirci (2002a). A survey on sensor networks. *Communications magazine, IEEE*, **40**(8), 102–114.
- Akyildiz, I.F., W. Su, Y. Sankarasubramaniam et E. Cayirci (2002b). Wireless sensor networks : a survey. *Computer networks*, **38**(4), 393–422.
- Al-Karaki, J.N. et A.E. Kamal (2004a). Routing techniques in wireless sensor networks : a survey. *Wireless Communications, IEEE*, **11**(6), 6–28.
- Al-Karaki, J.N. et A.E. Kamal (2004b). Routing techniques in wireless sensor networks : a survey. *Wireless Communications, IEEE*, **11**(6), 6–28.
- Anastasi, G., M. Conti, A. Passarella et L. Pelusi (2008). Mobile-relay forwarding in opportunistic networks. *Adaptation and Cross Layer Design in Wireless Networks*.
- Anastasi, G., M. Conti, M. Di Francesco et A. Passarella (2009). Energy conservation in wireless sensor networks : A survey. *Ad Hoc Networks*, **7**(3), 537–568.
- Arai, Y., T. Agui et M. Nakajima (1988). A fast DCT-SQ scheme for images. *Transactions of the Institute of Electronics and Communication Engineers of Japan. Section E*, **71**(11), 1095–1097.

- Arisha, K., M. Youssef et M. Younis (2002). Energy-aware tdma-based mac for sensor networks. *System-level power optimization for wireless multimedia communication*, pp. 21–40.
- Arora, A., P. Dutta, S. Bapat, V. Kulathumani, H. Zhang, V. Naik, V. Mittal, H. Cao, M. Demirbas, M. Gouda et al. (2004). A line in the sand : a wireless sensor network for target detection, classification, and tracking. *Computer Networks*, **46**(5), 605–634.
- Barakat, M. (2005). Smart dust : La nouvelle technologie.
- Becker, M., A.L. Beylot, R. Dhaou, A. Gupta, R. Kacimi et M. Marot (2009). Experimental study : Link quality and deployment issues in wireless sensor networks. *NETWORKING 2009*, pp. 14–25.
- Ben Saad, L., B. Tourancheau et al. (2010). Stratégie de placement des puits mobiles dans les réseaux de capteurs sans fil pour bâtiments.
- Bhardwaj, M., S. Soni et D.K. Kotary (2012). Comparative analysis of energy efficient routing protocol for wireless sensor network. *IJCA Special Issue on Wireless Communication and Mobile Networks*.
- Biagioni, E.S. et KW Bridges (2002). The application of remote sensor technology to assist the recovery of rare and endangered species. *International Journal of High Performance Computing Applications*, **16**(3), 315–324.
- Blough, D.M. et P. Santi (2002). Investigating upper bounds on network lifetime extension for cell-based energy conservation techniques in stationary ad hoc networks. In : *Proceedings of the 8th annual international conference on Mobile computing and networking*. ACM. pp. 183–192.
- Bo, X. et Z. Xuelong (1999). A global decision method for moving picture coding. *Consumer Electronics, IEEE Transactions on*, **45**(1), 84–90.
- Bodden, E., M. Clasen et J. Kneis (2007). Arithmetic coding revealed.
- Bracamonte, J., M. Ansorge et F. Pellandini (1996). VLSI systems for image compression : a power-consumption/image-resolution trade-off approach. In : *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*. Vol. 2952. pp. 591–596.
- Bracamonte, J., P. Stadelmann, M. Ansorge et F. Pellandini (2006). A multiplierless implementation scheme for the jpeg image coding algorithm.
- Bruekers, F. A. M. L. et A. W. M. van den Enden (1992). New networks for perfect inversion and perfect reconstruction. , **10**(1), 129–137.



- Bulusu, N., J. Heidemann et D. Estrin (2000). Gps-less low-cost outdoor localization for very small devices. *Personal Communications, IEEE*, **7**(5), 28–34.
- Carbunar, B., A. Grama, J. Vitek et O. Carbunar (2006). Redundancy and coverage detection in sensor networks. *ACM Transactions on Sensor Networks (TOSN)*, **2**(1), 94–128.
- Caruso, A., S. Chessa, S. De et A. Urpi (2005). Gps free coordinate assignment and routing in wireless sensor networks. In : *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*. Vol. 1. IEEE. pp. 150–160.
- Chang, C.C. et H. Aghajan (2006). Collaborative face orientation detection in wireless image sensor networks. In : *ACM SenSys Workshop on Distributed Smart Cameras (DSC)*.
- Chen, T.S., J. Chen et J.G. Chen (2003). A simple and efficient watermarking technique based on jpeg2000 codec. In : *Multimedia Software Engineering, 2003. Proceedings. Fifth International Symposium on*. IEEE. pp. 80–87.
- Chen, W.H., CH Smith et S. Fralick (1977). A fast computational algorithm for the discrete cosine transform. *Communications, IEEE Transactions on*, **25**(9), 1004–1009.
- Chiang, J.S., C.H. Chang, Y.S. Lin, C.Y. Hsieh et C.H. Hsia (2004). High-speed ebcot with dual context-modeling coding architecture for jpeg2000. In : *Circuits and Systems, 2004. ISCAS'04. Proceedings of the 2004 International Symposium on*. Vol. 3. IEEE. pp. III–865.
- Chiasserini, C.F. et E. Magli (2002). Energy consumption and image quality in wireless video-surveillance networks. In : *Personal, Indoor and Mobile Radio Communications, 2002. The 13th IEEE International Symposium on*. Vol. 5. IEEE. pp. 2357–2361.
- Chiasserini, C.F. et E. Magli (2004). Energy-efficient coding and error control for wireless video-surveillance networks. *Telecommunication Systems*, **26**(2), 369–387.
- Chiasserini, C.F., I. Chlamtac, P. Monti et A. Nucci (2006). Energy efficient design of wireless ad hoc networks. *NETWORKING 2002 : Networking Technologies, Services, and Protocols ; Performance of Computer and Communication Networks ; Mobile and Wireless Communications*, pp. 376–386.

- Christopoulos, CA, J. Bormans, J. Cornelis et AN Skodras (1995). The vector-radix fast cosine transform : Pruning and complexity analysis. *Signal processing*, **43**(2), 197–205.
- Crossbow, I. (n.d.). Technology mica2 : Wireless measurement system.
- Cui, J.H., J. Kong, M. Gerla et S. Zhou (2006). The challenges of building mobile underwater wireless networks for aquatic applications. *Network, IEEE*, **20**(3), 12–18.
- Cui, Y.L., Z.G. Chen et A.H. Chen (2012). Research and progress of image compression coding based on wavelet. *Advanced Materials Research*, **403**, 1352–1355.
- Dietrich, I. et F. Dressler (2009). On the lifetime of wireless sensor networks. *ACM Transactions on Sensor Networks (TOSN)*, **5**(1), 5.
- Doherty, L., L. El Ghaoui et al. (2001). Convex position estimation in wireless sensor networks. In : *INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*. Vol. 3. Ieee. pp. 1655–1663.
- Dong, B. (2009). A survey of underwater wireless sensor networks. In : *CAHSI 2009 ANNUAL MEETING*. p. 52.
- Duhamel, P. et H. H'Mida (1987). New 2ndct algorithms suitable for vlsi implementation. In : *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP'87..* Vol. 12. IEEE. pp. 1805–1808.
- Duran-Faundez, C. et V. Lecuire (2008). Error resilient image communication with chaotic pixel interleaving for wireless camera sensors. In : *Proceedings of the workshop on Real-world wireless sensor networks*. ACM. pp. 21–25.
- Duran-Faundez, C., V. Lecuire et F. Lepage (2011). Tiny block-size coding for energy-efficient image compression and communication in wireless camera sensor networks. *Signal Processing : Image Communication*.
- Egiazarian, K., J. Astola, N. Ponomarenko, V. Lukin, F. Battisti et M. Carli (2006). New full-reference quality metrics based on hvs. In : *CD-ROM Proceedings of the Second International Workshop on Video Processing and Quality Metrics*.
- Eskicioglu, A.M. et P.S. Fisher (1993). A survey of quality measures for gray scale image compression. In : *AIAA Computing in Aerospace Conference, 9 th, San Diego, CA*. pp. 304–313.
- Eskicioglu, A.M. et P.S. Fisher (1995). Image quality measures and their performance. *Communications, IEEE Transactions on*, **43**(12), 2959–2965.

- Estrin, D., L. Girod, G. Pottie et M. Srivastava (2001). Instrumenting the world with wireless sensor networks. In : *Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on*. Vol. 4. IEEE. pp. 2033–2036.
- Estrin, D., R. Govindan, J. Heidemann et S. Kumar (1999). Next century challenges : Scalable coordination in sensor networks. In : *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*. ACM. pp. 263–270.
- Feig, E. et E. Linzer (1992). The multiplicative complexity of discrete cosine transforms. *Advances in Applied Mathematics*, **13**(4), 494–503.
- Feig, E. et S. Winograd (1992). Fast algorithms for the discrete cosine transform. *Signal Processing, IEEE Transactions on*, **40**(9), 2174–2193.
- Ferrigno, L., S. Marano, V. Paciello et A. Pietrosanto (2005). Balancing computational and transmission power consumption in wireless image sensor networks. In : *Virtual Environments, Human-Computer Interfaces and Measurement Systems, 2005. VECIMS 2005. Proceedings of the 2005 IEEE International Conference on*. IEEE. pp. 6–pp.
- Fuller, S.H. et L.I. Millett (2011). Computing performance : Game over or next level?. *Computer*, **44**(1), 31–38.
- Gallager, R. et D. Van Voorhis (1975). Optimal source codes for geometrically distributed integer alphabets (corresp.). *Information Theory, IEEE Transactions on*, **21**(2), 228–230.
- Gallais, A., J. Carle, D. Simplot-Ryl et I. Stojmenovic (2008). Localized sensor area coverage with low communication overhead. *Mobile Computing, IEEE Transactions on*, **7**(5), 661–672.
- Gedik, B., L. Liu et P.S. Yu (2007). Asap : an adaptive sampling approach to data collection in sensor networks. *Parallel and Distributed Systems, IEEE Transactions on*, **18**(12), 1766–1783.
- Gersho, A. (1982). On the structure of vector quantizers. *Information Theory, IEEE Transactions on*, **28**(2), 157–166.
- Gersho, A. et R.M. Gray (1992). *Vector quantization and signal compression*. Springer Netherlands.

- Girod, B. et KW Stuhlmuller (1998). A content-dependent fast DCT for low bit-rate video coding. In : *Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on*. IEEE. pp. 80–84.
- Gobriel, S., D. Mosse et R. Cleric (2009). Tdma-asap : Sensor network tdma scheduling with adaptive slot-stealing and parallelism. In : *Distributed Computing Systems, 2009. ICDCS'09. 29th IEEE International Conference on*. Ieee. pp. 458–465.
- Goel, S., A. Passarella et T. Imielinski (2006). Using buddies to live longer in a boring world [sensor network protocol]. In : *Pervasive Computing and Communications Workshops, 2006. PerCom Workshops 2006. Fourth Annual IEEE International Conference on*. IEEE. pp. 5–pp.
- Goel, S. et T. Imielinski (2001). Prediction-based monitoring in sensor networks : taking lessons from mpeg. *ACM SIGCOMM Computer Communication Review*, **31**(5), 82–98.
- Golomb, SW (1966). Run-length encodings. *IEEE Transactions on Information Theory*, **12**, 399–401.
- Granata, J., M. Conner et R. Tolimieri (1992). The tensor product : a mathematical programming language for FFTs and other fast DSP operations. *Signal Processing Magazine, IEEE*, **9**(1), 40–48.
- Grgic, S., M. Grgic et B. Zovko-Cihlar (2001). Performance analysis of image compression using wavelets. *Industrial Electronics, IEEE Transactions on*, **48**(3), 682–695.
- Gutter, H. (1995). *La compression des images numériques*. Hermès.
- Haartsen, J.C. (2000). The bluetooth radio system. *Personal Communications, IEEE*, **7**(1), 28–36.
- Halkes, G. et K. Langendoen (2007). Crankshaft : An energy-efficient mac-protocol for dense wireless sensor networks. *Wireless Sensor Networks*, pp. 228–244.
- Heinzelman, W.B., A.P. Chandrakasan et H. Balakrishnan (2002). An application-specific protocol architecture for wireless microsensor networks. *Wireless Communications, IEEE Transactions on*, **1**(4), 660–670.
- Heinzelman, W.R., A. Chandrakasan et H. Balakrishnan (2000). Energy-efficient communication protocol for wireless microsensor networks. In : *System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference on*. IEEE. pp. 10–pp.

- Hicks, J., J. Paek, S. Coe, R. Govindan et D. Estrin (2008). An easily deployable wireless imaging system. *Proc of ImageSense*.
- Howard, P.G. et J.S. Vitter (1992). Practical implementations of arithmetic coding. *Image and text compression*, pp. 85–112.
- Huffman, D.A. (1952). A method for the construction of minimum-redundancy codes. *Proceedings of the IRE*, **40**(9), 1098–1101.
- Ingelrest, F., D. Simplot-Ryl et I. Stojmenovic (2006). Optimal transmission radius for energy efficient broadcasting protocols in ad hoc and sensor networks. *Parallel and Distributed Systems, IEEE Transactions on*, **17**(6), 536–547.
- Intanagonwiwat, C., R. Govindan, D. Estrin, J. Heidemann et F. Silva (2003). Directed diffusion for wireless sensor networking. *Networking, IEEE/ACM Transactions on*, **11**(1), 2–16.
- Jatlaoui, M.M. (2009). Capteurs passifs à transduction électromagnétique pour la mesure sans fil de la pression.
- Jeong, H., J. Kim et W. Cho (2004). Low-power multiplierless DCT architecture using image correlation. *Consumer Electronics, IEEE Transactions on*, **50**(1), 262–267.
- Jin, K., Y. Zhang et D. Tian (2012). Based on the improvement of leach protocol for wireless sensor network routing algorithm. In : *2012 International Conference on Intelligent System Design and Engineering Application*. IEEE. pp. 1525–1528.
- Jones, C.E., K.M. Sivalingam, P. Agrawal et J.C. Chen (2001). A survey of energy efficient network protocols for wireless networks. *wireless networks*, **7**(4), 343–358.
- Jung, J., M. Antonini et M. Barlaud (1998). Optimal JPEG decoding. In : *Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on*. Vol. 1. IEEE. pp. 410–414.
- Jung, S.H., S.K. Mitra et D. Mukherjee (1996). Subband DCT : Definition, analysis, and applications. *Circuits and Systems for Video Technology, IEEE Transactions on*, **6**(3), 273–286.
- Kaddachi, M.L., A. Soudani, V. Lecuire, K. Torki, L. Makkaoui et J.M. Moureaux (2011). Low power hardware-based image compression solution for wireless camera sensor networks. *Computer Standards & Interfaces*.
- Kaddachi, Mohamed Lassaad (2012). Transfert des images et gestion de la qualité du service dans les réseaux de capteurs sans fils. Master's thesis. FACULTE DES SCIENCES DE MONASTIR.

- Karl, H. et A. Willig (2005). Protocols and architectures for wireless sensor networks. *booksgooglecom*.
- Karl, H. et A. Willig (2007). *Protocols and architectures for wireless sensor networks*. Wiley-Interscience.
- Kim, K.L. et S.W. Ra (2004). Performance improvement of the split coder. *Signal Processing : Image Communication*, **19**(1), 29–36.
- Kleinrock, L. et F. Tobagi (1975). Packet switching in radio channels : Part i—carrier sense multiple-access modes and their throughput-delay characteristics. *Communications, IEEE Transactions on*, **23**(12), 1400–1416.
- Kreyszig, E. (2007). *Advanced engineering mathematics*. Wiley-India.
- Krishnamachari, B., D. Estrin et S. Wicker (2002). Modelling data-centric routing in wireless sensor networks. In : *IEEE infocom*. Vol. 2. pp. 1–11.
- Labrador, M.A. et P.M. Wightman (2009). *Topology Control in Wireless Sensor Networks : with a companion simulation tool for teaching and research*. Springer Verlag.
- Le Callet, P. (2001). Critères objectifs avec référence de qualité visuelle des images couleur. PhD thesis.
- Le Pennec, E. et S. Mallat (2005). Sparse geometric image representations with bandelets. *Image Processing, IEEE Transactions on*, **14**(4), 423–438.
- Lecuire, V., C. Duran-Faundez et N. Krommenacker (2008). Energy-efficient image transmission in sensor networks. *International Journal of Sensor Networks*, **4**(1), 37–47.
- Lee, D.U., H. Kim, S. Tu, M. Rahimi, D. Estrin et J.D. Villasenor (2007). Energy-optimized image communication on resource-constrained sensor platforms. In : *Information Processing in Sensor Networks, 2007. IPSN 2007. 6th International Symposium on*. IEEE. pp. 216–225.
- Lee, H. et H. Aghajan (2006). Vision-enabled node localization in wireless sensor networks. *COGNITIVE systems with Interactive Sensors (COGIS)*.
- Lengwehasatit, K. et A. Ortega (2004). Scalable variable complexity approximate forward dct. *Circuits and Systems for Video Technology, IEEE Transactions on*, **14**(11), 1236–1248.
- Levis, P., S. Madden, J. Polastre, R. Szewczyk, K. Whitehouse, A. Woo, D. Gay, J. Hill, M. Welsh, E. Brewer et al. (2005). Tinyos : An operating system for sensor networks. *Ambient intelligence*.

- Li, J. et P. Mohapatra (2005). An analytical model for the energy hole problem in many-to-one sensor networks. In : *IEEE Vehicular Technology Conference*. Vol. 62. IEEE ; 1999. p. 2721.
- Li, J. et P. Mohapatra (2007). Analytical modeling and mitigation techniques for the energy hole problem in sensor networks. *Pervasive and Mobile Computing*, **3**(3), 233–254.
- Li, N. et J.C. Hou (2004). Topology control in heterogeneous wireless networks : Problems and solutions. In : *INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*. Vol. 1. IEEE.
- Lian, C.J., K.F. Chen, H.H. Chen et L.G. Chen (2003). Analysis and architecture design of block-coding engine for ebcot in jpeg 2000. *Circuits and Systems for Video Technology, IEEE Transactions on*, **13**(3), 219–230.
- Lian, J., K. Naik et G.B. Agnew (2006). Data capacity improvement of wireless sensor networks using non-uniform sensor distribution. *International Journal of Distributed Sensor Networks*, **2**(2), 121–145.
- Liang, J. et T.D. Tran (2000). Approximating the DCT with the lifting scheme : systematic design and applications. In : *Signals, Systems and Computers, 2000. Conference Record of the Thirty-Fourth Asilomar Conference on*. Vol. 1. IEEE. pp. 192–196.
- Linde, Y., A. Buzo et R. Gray (1980). An algorithm for vector quantizer design. *Communications, IEEE Transactions on*, **28**(1), 84–95.
- Lindsey, S. et C.S. Raghavendra (2002). Pegasus : Power-efficient gathering in sensor information systems. In : *Aerospace Conference Proceedings, 2002. IEEE*. Vol. 3. IEEE. pp. 3–1125.
- Loeffler, C., A. Lieenberg et G.S. Moschytz (n.d.). PRACTICAL FAST 1-D DCT ALGORITHMS WITH 11 MULTIPLICATIONS. *Variations*, **3**, 7.
- Loeffler, C.L. et A. Moschytz (1989). Practical fast 1-D DCT algorithms with 11 multiplications. In : *1989 International Conference on Acoustics, Speech, and Signal Processing, 1989. ICASSP-89..* pp. 988–991.
- Lu, G., B. Krishnamachari et C.S. Raghavendra (2004). An adaptive energy-efficient and low-latency mac for data gathering in wireless sensor networks. In : *Parallel and Distributed Processing Symposium, 2004. Proceedings. 18th International. Ieee*. p. 224.

- Lu, Q., W. Luo, J. Wang et B. Chen (2008). Low-complexity and energy efficient image compression scheme for wireless sensor networks. *Computer Networks*, **52**(13), 2594–2603.
- Mahfoudh, S. et P. Minet (2008). Survey of energy efficient strategies in wireless ad hoc and sensor networks. In : *Networking, 2008. ICN 2008. Seventh International Conference on*. IEEE. pp. 1–7.
- Maimour, M. (2007). Multipath routing protocol for layered video transport in wireless sensor networks.
- Makkaoui, L., V. Lecuire et J.M. Moureaux (2010). Fast zonal DCT-based image compression for Wireless Camera Sensor Networks. In *Image Processing Theory Tools and Applications (IPTA), 2010 2nd International Conference on*.
- Mallat, S.G. (1999). *A wavelet tour of signal processing*. Academic Pr.
- Malvar, H.S. (2006). Adaptive run-length/golomb-rice encoding of quantized generalized gaussian sources with unknown statistics. In : *Data Compression Conference, 2006. DCC 2006. Proceedings*. IEEE. pp. 23–32.
- Mammeri, A., A. Khoumsi, D. Ziou et B. Hadjou (2008). Modeling and adapting JPEG to the energy requirements of VSN. In : *Computer Communications and Networks, 2008. ICCCN'08. Proceedings of 17th International Conference on*. IEEE. pp. 1–6.
- Manjeshwar, A. et D.P. Agrawal (2001). Teen : a routing protocol for enhanced efficiency in wireless sensor networks. In : *Parallel and Distributed Processing Symposium., Proceedings 15th International*. IEEE. pp. 2009–2015.
- Marcelloni, F. et M. Vecchio (2008). A simple algorithm for data compression in wireless sensor networks. *Communications Letters, IEEE*, **12**(6), 411–413.
- Marino, F., V. Piuri et E.E. Swartzlander Jr (1999). A parallel implementation of the 2-d discrete wavelet transform without interprocessor communications. *Signal Processing, IEEE Transactions on*, **47**(11), 3179–3184.
- Marpe, D., H. Schwarz et T. Wiegand (2003). Context-based adaptive binary arithmetic coding in the h. 264/avc video compression standard. *Circuits and Systems for Video Technology, IEEE Transactions on*, **13**(7), 620–636.
- Meng-Chun, L., D. Lan-Rong et W. Ping-Kuo (2006). An ultra-low-power image compressor for capsule endoscope. *BioMedical Engineering OnLine*.



- Min, B., S. Yoon, J. Ra et D.S. Park (2007). Enhanced renormalization algorithm in mqc-coder of jpeg2000. In : *Information Technology Convergence, 2007. ISITC 2007. International Symposium on*. IEEE. pp. 213–216.
- Minami, S. et A. Zakhor (1995). An optimization approach for removing blocking effects in transform coding. *Circuits and Systems for Video Technology, IEEE Transactions on*, **5**(2), 74–82.
- MIT (2003). Emerging technologies that will change the world. *MIT Technology Review*, **106**(1), 10–51.
- Moccagatta, I. et M.Z. Coban (2000). Computational complexity evaluation of jpeg 2000. In : *Proceedings of SPIE*. Vol. 4115. p. 521.
- Moureaux, J.M. (2007). Quantification vectorielle algébrique : un outil performant pour la compression et le tatouage d’images fixes. *HDR, Université Henri Poincaré, Nancy*.
- Moureaux, J.M., P. Loyer et M. Antonini (1998). Low-complexity indexing method for zn and dn lattice quantizers. *Communications, IEEE Transactions on*, **46**(12), 1602–1609.
- Nelson, M. et H. Soulard (1993). *La compression des données : texte, images, sons*. Dunod.
- Nikolaidis, A. (2012). Local distortion resistant image watermarking re-lying on salient feature extraction. *EURASIP Journal on Advances in Signal Processing*, **2012**(1), 97.
- Ninassi, A. (2009). De la perception locale des distorsions de codage à l’appréciation globale de la qualité visuelle des images et vidéos. Apport de l’attention visuelle dans le jugement de qualité.. PhD thesis. Université de Nantes.
- Nyeem, H., W. Boles et C. Boyd (2012). On the robustness and security of digital image watermarking. In : *Proceedings of International Conference on Informatics, Electronics & Vision*.
- Oto, E., F. Lau et H. Aghajan (2006). Color-based multiple agent tracking for wireless image sensor networks. In : *Advanced Concepts for Intelligent Vision Systems*. Springer. pp. 299–310.
- Pao, I.M. et M.T. Sun (1999). Modeling dct coefficients for fast video encoding. *Circuits and Systems for Video Technology, IEEE Transactions on*, **9**(4), 608–616.
- Papadonikolakis, M., V. Pantazis et A.P. Kakarountas (2007). Efficient high-performance asic implementation of jpeg-ls encoder. In : *Design, Automation & Test in Europe Conference & Exhibition, 2007. DATE’07*. IEEE. pp. 1–6.

- Parisis, A., P. Carré, C. Fernandez-Maloigne et N. Laurent (2004). Tatouage d'images couleur avec adaptation locale des forces de marquage. *Compression et Représentation des Signaux Audiovisuels (CORESA 2004)*.
- Park, M.K. et V. Rodoplu (2007). Uwan-mac : An energy-efficient mac protocol for underwater acoustic wireless sensor networks. *Oceanic Engineering, IEEE Journal of*, **32**(3), 710–720.
- Pedersen, M. et J.Y. Hardeberg (2009). Survey of full-reference image quality metrics.
- Perillo, M. et W. Heinzelman (2004). Wireless sensor network protocols. *Algorithms and Protocols for Wireless and Mobile Networks, Eds. A. Boukerche et al., CRC Hall Publishers*.
- Petrovic, D., R.C. Shah, K. Ramchandran et J. Rabaey (2003). Data funneling : Routing with aggregation and compression for wireless sensor networks. In : *Sensor Network Protocols and Applications, 2003. Proceedings of the First IEEE. 2003 IEEE International Workshop on*. IEEE. pp. 156–162.
- Pister, Kristofer S.J. (1997). Smart dust : Baa97-43 proposal abstract,.
- Polastre, J., R. Szewczyk et D. Culler (2005). Telos : enabling ultra-low power wireless research. In : *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on*. Ieee. pp. 364–369.
- Ponomarenko, N., F. Silvestri, K. Egiazarian, M. Carli, J. Astola et V. Lukin (2007). On between-coefficient contrast masking of dct basis functions. In : *N. Ponomarenko, F. Silvestri, K. Egiazarian, M. Carli, J. Astola, V. Lukin/Proceedings of the Third International Workshop on Video Processing and Quality Metrics, Scottsdale, Arizona, USA*. p. 4.
- Pottie, G.J. et W.J. Kaiser (2000). Wireless integrated network sensors. *Communications of the ACM*, **43**(5), 51–58.
- Pradhan, S.S., J. Kusuma et K. Ramchandran (2002). Distributed compression in a dense microsensor network. *Signal Processing Magazine, IEEE*, **19**(2), 51–60.
- Qi, H., P.T. Kuruganti et Y. Xu (2002). The development of localized algorithms in wireless sensor networks. *Sensors*, **2**(7), 286–293.
- Quails, R.J., J.M. Scott et W.B. DeOreo (2001). Soil moisture sensors for urban landscape irrigation : Effectweness and reliability<sup>1</sup>. *JAWRA Journal of the American Water Resources Association*, **37**(3), 547–559.

- Raghunathan, V., C. Schurgers, S. Park et M.B. Srivastava (2002). Energy-aware wireless microsensor networks. *Signal Processing Magazine, IEEE*, **19**(2), 40–50.
- Rahimi, M., R. Baer, O.I. Iroezi, J.C. Garcia, J. Warrior, D. Estrin et M. Srivastava (2005). Cyclops : in situ image sensing and interpretation in wireless sensor networks. In : *Proceedings of the 3rd international conference on Embedded networked sensor systems*. ACM. pp. 192–204.
- Rajendran, V., JJ Garcia-Luna-Aveces et K. Obraczka (2005). Energy-efficient, application-aware medium access for sensor networks. In : *Mobile Adhoc and Sensor Systems Conference, 2005. IEEE International Conference on*. IEEE. pp. 8–pp.
- Rajendran, V., K. Obraczka et J.J. Garcia-Luna-Aceves (2006). Energy-efficient, collision-free medium access control for wireless sensor networks. *Wireless Networks*, **12**(1), 63–78.
- Rein, S. et M. Reisslein (2011). Low-memory wavelet transforms for wireless sensor networks : a tutorial. *Communications Surveys & Tutorials, IEEE*, (99), 1–17.
- Rhee, I., A. Warriier, M. Aia, J. Min et M.L. Sichitiu (2008). Z-mac : a hybrid mac for wireless sensor networks. *IEEE/ACM Transactions on Networking (TON)*, **16**(3), 511–524.
- Rice, R. et J. Plaunt (1971). Adaptive variable-length coding for efficient compression of spacecraft television data. *Communication Technology, IEEE Transactions on*, **19**(6), 889–897.
- Royer, E.M. et C.K. Toh (1999). A review of current routing protocols for ad hoc mobile wireless networks. *Personal Communications, IEEE*, **6**(2), 46–55.
- Sankarasubramaniam, Y., I.F. Akyildiz et SW McLaughlin (2003). Energy efficiency based packet size optimization in wireless sensor networks. In : *Sensor Network Protocols and Applications, 2003. Proceedings of the First IEEE. 2003 IEEE International Workshop on*. IEEE. pp. 1–8.
- Santa-Cruz, D. et T. Ebrahimi (2000). A study of jpeg 2000 still image coding versus other standards. In : *Proc. of the X European Signal Processing Conference*. Vol. 2. pp. 673–676.
- Santoso, S., E.J. Powers et WM Grady (1997). Power quality disturbance data compression using wavelet transform methods. *Power Delivery, IEEE Transactions on*, **12**(3), 1250–1257.

- Savvides, A. et M.B. Srivastava (2002). A distributed computation platform for wireless embedded sensing. In : *Computer Design : VLSI in Computers and Processors, 2002. Proceedings. 2002 IEEE International Conference on*. IEEE. pp. 220–225.
- Sazzad, Z.M.P., Y. Kawayoke et Y. Horita (2007). Spatial features based no reference image quality assessment for jpeg2000. In : *Image Processing, 2007. ICIP 2007. IEEE International Conference on*. Vol. 3. IEEE. pp. III–517.
- Shih, E., S.H. Cho, N. Ickes, R. Min, A. Sinha, A. Wang et A. Chandrakasan (2001). Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks. In : *Proceedings of the 7th annual international conference on Mobile computing and networking*. ACM. pp. 272–287.
- Skodras, A.N. (1994). Fast discrete cosine transform pruning. *Signal Processing, IEEE Transactions on*, **42**(7), 1833–1837.
- Soro, S. et W. Heinzelman (2009). A survey of visual sensor networks. *Advances in Multimedia*.
- Soro, S. et W.B. Heinzelman (2005). Prolonging the lifetime of wireless sensor networks via unequal clustering. In : *Parallel and Distributed Processing Symposium, 2005. Proceedings. 19th IEEE International*. IEEE. pp. 8–pp.
- Starner, T.E. (2003). Powerful change part 1 : batteries and possible alternatives for the mobile market. *Pervasive Computing, IEEE*, **2**(4), 86–88.
- Subramanian, N.V. (2004). Survey on energy-aware routing and routing protocols for sensor networks. Technical report. Technical Report, Computer Science, University of North Carolina, Charlotte.
- Sun, C.C., S.J. Ruan, B. Heyne et J. Goetze (2007). Low-power and high-quality Cordic-based Loeffler DCT for signal processing. *Circuits, Devices & Systems, IET*, **1**(6), 453–461.
- Sun, S.H. et S.J. Lee (2003). A jpeg chip for image compression and decompression. *The Journal of VLSI Signal Processing*, **35**(1), 43–60.
- Taubman, D.S., M.W. Marcellin et M. Rabbani (2002). Jpeg2000 : Image compression fundamentals, standards and practice. *Journal of Electronic Imaging*, **11**, 286.
- Taylor, C.J. (2004). A scheme for calibrating smart camera networks using active lights. In : *Proceedings of the 2nd international conference on Embedded networked sensor systems*. ACM. pp. 322–322.

- Taylor, C.N. et S. Dey (2001). Adaptive image compression for wireless multimedia communication. In : *Communications, 2001. ICC 2001. IEEE International Conference on*. Vol. 6. IEEE. pp. 1925–1929.
- Thirumalai, V., I. Tosic et P. Frossard (2008). Symmetric distributed coding of stereo omnidirectional images. *Signal Processing : Image Communication*, **23**(5), 379–390.
- Thorn, J. (2005). Deciphering tinyos serial packets. *Octave Tech Brief*, pp. 5–01.
- Tian, D. et N.D. Georganas (2002). A coverage-preserving node scheduling scheme for large wireless sensor networks. In : *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*. ACM. pp. 32–41.
- Tian, Z., D. Yuan et Q. Liang (2008). Energy efficiency analysis of error control schemes in wireless sensor networks. In : *Wireless Communications and Mobile Computing Conference, 2008. IWCMC'08. International*. IEEE. pp. 401–405.
- Tilak, S., N.B. Abu-Ghazaleh et W. Heinzelman (2002). A taxonomy of wireless micro-sensor network models. *ACM SIGMOBILE Mobile Computing and Communications Review*, **6**(2), 28–36.
- Tran, T.D. (1999). A fast multiplierless block transform for image and video compression. In : *Image Processing, 1999. ICIP 99. Proceedings. 1999 International Conference on*. Vol. 3. IEEE. pp. 822–826.
- Tran, T.D. (2000). The BinDCT : Fast multiplierless approximation of the DCT. *Signal Processing Letters, IEEE*, **7**(6), 141–144.
- Vaidyanathan, PP (1990). Multirate digital filters, filter banks, polyphase networks, and applications : A tutorial. *Proceedings of the IEEE*, **78**(1), 56–93.
- Van Dam, T. et K. Langendoen (2003). An adaptive energy-efficient mac protocol for wireless sensor networks. In : *Proceedings of the 1st international conference on Embedded networked sensor systems*. ACM. pp. 171–180.
- Veres, A., A.T. Campbell, M. Barry et L.H. Sun (2001). Supporting service differentiation in wireless packet networks using distributed control. *Selected Areas in Communications, IEEE Journal on*, **19**(10), 2081–2093.
- Volder, J. (1956). Binary computation algorithm for coordinate rotation and function generation.. Technical report. i.
- Vuran, M.C. et I.F. Akyildiz (2006). Spatial correlation-based collaborative medium access control in wireless sensor networks. *Networking, IEEE/ACM Transactions on*, **14**(2), 316–329.

- Wagner, R., R. Nowak et R. Baraniuk (2003). Distributed image compression for sensor networks using correspondence analysis and super-resolution. In : *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*. Vol. 1. IEEE. pp. I–597.
- Wang, Z. (1991). Pruning the fast discrete cosine transform. *Communications, IEEE Transactions on*, **39**(5), 640–643.
- Wang, Z., A.C. Bovik et E.P. Simoncelli (2005). Structural approaches to image quality assessment. *Handbook of Image and Video Processing*, **7**, 18.
- Wang, Z., A.C. Bovik, H.R. Sheikh et E.P. Simoncelli (2004). Image quality assessment : From error visibility to structural similarity. *Image Processing, IEEE Transactions on*, **13**(4), 600–612.
- Wang, Z., E.P. Simoncelli et A.C. Bovik (2003). Multiscale structural similarity for image quality assessment. In : *Signals, Systems and Computers, 2003. Conference Record of the Thirty-Seventh Asilomar Conference on*. Vol. 2. IEEE. pp. 1398–1402.
- Wang, Z. et A.C. Bovik (2002). A universal image quality index. *Signal Processing Letters, IEEE*, **9**(3), 81–84.
- Warneke, B., M. Last, B. Liebowitz et K.S.J. Pister (2001). Smart dust : Communicating with a cubic-millimeter computer. *Computer*, **34**(1), 44–51.
- Wawerla, J., S. Marshall, G. Mori, K. Rothley et P. Sabzmeydani (2009). Bearcam : Automated wildlife monitoring at the arctic circle. *Machine Vision and Applications*, **20**(5), 303–317.
- Weinberger, M.J., G. Seroussi et G. Sapiro (1996). Loco-i : A low complexity, context-based, lossless image compression algorithm. In : *Data Compression Conference, 1996. DCC'96. Proceedings*. IEEE. pp. 140–149.
- Witten, I.H., A. Moffat et T.C. Bell (1999). *Managing gigabytes : compressing and indexing documents and images*. Morgan Kaufmann.
- Witten, I.H., R.M. Neal et J.G. Cleary (1987). Arithmetic coding for data compression. *Communications of the ACM*, **30**(6), 520–540.
- Wu, H. et A.A. Abouzeid (2004a). Energy efficient distributed jpeg2000 image compression in multihop wireless networks. In : *Proc. of IEEE Workshop on Applications and Services in Wireless Networks*. pp. 152–160.

- Wu, H. et A.A. Abouzeid (2004b). Power aware image transmission in energy constrained wireless networks. In : *Computers and Communications, 2004. Proceedings. ISCC 2004. Ninth International Symposium on*. Vol. 1. IEEE. pp. 202–207.
- Xu, Y., J. Heidemann et D. Estrin (2001). Geography-informed energy conservation for ad hoc routing. In : *Proceedings of the 7th annual international conference on Mobile computing and networking*. ACM. pp. 70–84.
- Yao, Y. et J. Gehrke (2002). The cougar approach to in-network query processing in sensor networks. *ACM Sigmod Record*, **31**(3), 9–18.
- Ye, W., J. Heidemann et D. Estrin (2002). An energy-efficient mac protocol for wireless sensor networks. In : *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*. Vol. 3. IEEE. pp. 1567–1576.
- Ye, W., J. Heidemann et D. Estrin (2004). Medium access control with coordinated adaptive sleeping for wireless sensor networks. *Networking, IEEE/ACM Transactions on*, **12**(3), 493–506.
- Yu, Y., R. Govindan et D. Estrin (2001). Geographical and energy aware routing : A recursive data dissemination protocol for wireless sensor networks. Technical report. Technical report ucla/csd-tr-01-0023, UCLA Computer Science Department.
- Zhou, R., L. Liu, S. Yin, A. Luo, X. Chen et S. Wei (2010). A vlsi design of sensor node for wireless image sensor network. In : *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on*. IEEE. pp. 149–152.











---

## Résumé

Les réseaux de capteurs sans fil d'images sont utilisés aujourd'hui dans de nombreuses applications qui diffèrent par leurs objectifs et leurs contraintes individuelles. Toutefois, le dénominateur commun de toutes les applications de réseaux de capteurs d'images reste la vulnérabilité des noeuds-capteurs en raison de leurs ressources matérielles limitées dont la plus contraignante est l'énergie. En effet, les technologies sans fil disponibles dans ce type de réseaux sont généralement à faible portée, et les ressources matérielles (CPU, batterie) sont également de faible puissance. Il faut donc répondre à un double objectif : l'efficacité d'une solution tout en offrant une bonne qualité d'image à la réception.

La contribution de cette thèse porte principalement sur l'étude des méthodes de traitement et de compression d'images au noeud-caméra, nous avons proposé une nouvelle méthode de compression d'images qui permet d'améliorer l'efficacité énergétique des réseaux de capteurs d'images. Des expérimentations sur une plate-forme réelle de réseau de capteurs d'images ont été réalisées afin de démontrer la validité de nos propositions, en mesurant des aspects telles que la quantité de mémoire requise pour l'implantation logicielle de nos algorithmes, leur consommation d'énergie et leur temps d'exécution.

Nous présentons aussi, les résultats de synthèse de la chaîne de compression proposée sur des systèmes à puce FPGA et ASIC.

## Mots clés

Réseaux de capteurs sans fil, capteur d'image, compression d'images, conservation de l'énergie, transformée en cosinus discrete zonale.

## Abstract

The increasing development of Wireless Camera Sensor Networks today allows a wide variety of applications with different objectives and constraints. However, the common problem of all the applications of sensor networks remains the vulnerability of sensors nodes because of their limitation in hardware resources, the most restricting being energy. Indeed, the available wireless technologies in this type of networks are usually a low-power, short-range wireless technology and low power hardware resources (CPU, battery). So we should meet a twofold objective : an efficient solution while delivering outstanding image quality on reception.

This thesis concentrates mainly on the study and evaluation of compression methods dedicated to transmission over wireless camera sensor networks. We have suggested a new image compression method which decreases the energy consumption of sensors and thus maintains a long network lifetime. We evaluate its hardware implementation using experiments on real camera sensor platforms in order to show the validity of our propositions, by measuring aspects such as the quantity of memory required for the implementation program of our algorithms, the energy consumption and the execution time.

We then focus on the study of the hardware features of our proposed method of synthesis of the compression circuit when implemented on a FPGA and ASIC chip prototype.

## Keywords

wireless sensor network, camera sensor, image compression, energy consumption, Zonal discrete cosine transform