

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

*Ministère de l'enseignement supérieur  
et de la recherche scientifique*

INSTITUT NATIONAL DE FORMATION EN INFORMATIQUE  
OUED-SMAR

*MEMOIRE DE FIN D'ETUDE*

*POUR L'OBTENTION DU DIPLOME  
D'INGENIEUR D'ETAT EN INFORMATIQUE*

*Option*

**SYSTEMES INFORMATIQUES**

*THEME*

**COMPRESSION D'IMAGES EN NIVEAUX DE GRIS  
PAR L'APPROCHE WSQ  
SOUS LINUX**

**Organisme d'accueil : INI \_CERIST**

**Encadré par :**

M<sup>lle</sup> B. Oussena

**Réalisé par :**

Amira Nasreddine

***PROMOTION 2000-2001***



## ***Résumé***

*L'objectif principal de ce travail consiste à implémenter un compresseur d'images en niveaux de gris sous linux à l'aide de la transformée en ondelettes, basé sur l'algorithme de S.Mallat et l'approche WSQ dont le but est de réduire la complexité du codeur Huffman.*

**Mot clés** : *Compression, Image, Linux, La Transformée en ondelette, WSQ, HUFFMAN*

# SOMMAIRES

---

---

## Chapitre I : Concepts de base

I. INTRODUCTION	1
II. GENERALITES ET DEFINITION	2
II.1. Définition d'une image	2
II.2. La numérisation	2
II.3. Pixel	3
II.4. Niveau de gris	3
II.5. Bitmap	3
II.6. Opération de filtrage numérique	4
II.6.1. Filtre passe-haut	4
II.6.2. Filtre passe-bas	4
II.6.3. Filtre passe-bande	4
III. GENERALITES SUR LA COMPRESSION D'IMAGES	5
III.1. Principe de la compression d'image	5
III.1.1. Décorrélacion	5
III.1.2. Quantification	6
III.1.3. Codage	6
III.2. Les différents critères de compression	6
III.2.1. Compression symétrique et compression asymétrique	6
III.2.2. Compression conservative et non conservative	6
III.2.3. Compression non adaptatif, adaptatif et semi-adaptatif	6
III.3. Critères de choix d'une méthode de compression	7
III.4. Quelques méthodes de compression	7
III.4.1. Transformée	7
III.4.2. Quantification vectoriel	7
III.4.3. Transformation Pyramidales	8
III.5. Les différentes méthodes de codage	8
III.5.1. Codage de Huffman	8
III.5.2. Codage dynamique	11
III.5.3. Codage a préfixe	12
III.5.4. Codage arithmétique	13
III.5.5. Codage RLE (de répétition)	15
III.5.6. Codage par quadtree	16
III.5.7. Codage par le bintree	17
III.6. Domaine d'utilisation	19
III.6.1. Transmission	19

III.6.2. Stockage	19
III.7. Objectifs	19
IV. CONCLUSION	20

## Chapitre II : La transformée en ondelettes

I. INTRODUCTION	21
II. REMARQUES SUR LA TRANSFORMEE DE FOURRIER	22
III. C'EST QUOI UNE ONDELETTE ?	23
III.1. PROPRIETES	23
III.2. Famille d'ondelettes	24
III.3. La base d'une ondelette	25
IV. LA TRANSFORMEE EN ONDELETTES CONTINUE(CWT)	25
V. LA TRANSFORMEE EN ONDELETTES DISCRETE(DWT)	26
V.1. Définition	27
V.2. Analyse Multi-résolution	27
V.3. Construction d'une base orthonormale à partir de l'analyse multi-résolution	29
V.3.1. Unidimensionnelle	29
V.3.2. Bidimensionnelle	30
V.4. Implémentation	33
V.4.1. Algorithme unidimensionnel	33
V.4.2. Algorithme Bidimensionnelle	34
VI. QUELQUES TYPES D'ONDELETTES	36
VI.1. Ondelettes continues	36
VI.1.1. L'ondelette de Morlet	36
VI.1.2. Le chapeau mexicain	37
VI.2. Ondelette orthonormale	37
VI.2.1. L'ondelette de Littewood-Paley	38
VI.2.2. Les ondelettes de Daubechies	39
VI.3. Les ondelettes biorthogonales	40
VII. CONCLUSION	40

## Chapitre III : Environnement de travail

### A. UNIX /LINUX

I. HISTORIQUE	42
II. UNIX	43
III. LINUX	44
III.1. Fonctionnalités de Linux	44
III.1.1. Le noyau	44
III.1.2. Les applications	45

III.1.3. Compatibilité avec d'autres systèmes	45
III.2. Caractéristiques	46
III.3. Quelle est alors la place de linux ?	48
IV. DEMARRAGE DE LINUX	48
IV.1. Système commander	48
IV.2. Programme lilo	48
V. LE SHELL	49
VI. PROCESSUS	50
VI.1. Création des processus	50
VI.2. Destruction de processus	50
VI.3. les primitives d'entrées sorties	51
VII. L'INTERFACE AVEC LES TERMINAUX	51
VII.1. le mode de la liaison à un terminal	52
VII.2. Premier plan/Arrière plan	52
VII.3. Mode canonique /non canonique	53
VII.3.1. Mode canonique	53
VII.3.2. Mode non canonique	53
VII.1.A. GESTIONS DES EVENEMENTS	54
VIII.1. Opérations bloquantes/Opérations non bloquantes	54
VIII.2. Les signaux	54
<b>B. TOOLKIT GUI</b>	55
I. FRAMEWORK QT	55
II. PORTABILITE	56
III. QT, POUR QUOI FAIRE ?	57
IV. LES SIGNAUX ET LES SLOTS	57
V. UTILITAIRE MAKEFILE	58
<b>C. LES VERSIONS DE LINUX</b>	59

## Chapitre IV : Conception & Implémentation

I. INTRODUCTION	60
II. ARCHITECTURE DU LOGICIEL	61
III. ARCHITECTURE GENERALE DE LA METHODE WSQ	62
III.1. Transformée en ondelettes discrète (DWT)	62
III.1.1. Adaptation des filtres	62
III.1.2. Convolution	63
III.1.3. Décimation	65
III.2. Transformée en ondelettes discrète inverse (IDWT)	70
III.2.1. Adaptation des filtres	70
III.2.2. Interpolation	70
III.2.3. Convolution	70
III.3. La quantification	73

III.3.1. Choix de la largeur du zéro $Z_k$	73
III.3.2. Choix de la largeur de quantification $Q_k$	73
III.3.3. Paramètres de quantification	73
III.4. La déquantification	74
III.5. Codage	75
III.5.1. Phase lecture du fichier	75
III.5.2. Phase création d'arbre de Huffman et génération des codes	78
III.5.3. Phase écriture fichier	79
III.6. Décodage	80
IV. CODAGE ADAPTATIF	81
V. CONCLUSION	86

## Chapitre V : Tests & Résultats

I. INTRODUCTION	87
II. PARAMETRES	88
III. TESTS	88
III.1. Effet de la longueur du filtre	89
III.1.1. Codage <i>WSQ_Huffman</i>	89
III.1.2. Codage <i>WSQ_Huffman_Adaptatif</i>	90
III.2. Effet de la qualité de l'image	92
III.2.1. Codage <i>WSQ_Huffman</i>	92
III.2.2. Codage <i>WSQ_Huffman_Adaptatif</i>	93
III.3. Effet de la décomposition	96
IV. CONCLUSION	97
CONCLUSION GENERALE	98
<b>ANNEXE A</b>	100
<b>ANNEXE B</b>	101