



République Algérienne Démocratique et Populaire

Ministère de l'enseignement supérieur et de la recherche
scientifique



Université Kasdi Merbah –Ouargla- Algérie

Faculté des Nouvelles Technologies de l'Information et la Communication

Département de l'informatique et Technologies de l'information

DOMAINE : Mathématique et Informatique

FILIERE : Informatique

SPECIALITE : Administration et Sécurité des Réseaux

MÉMOIRE DE MASTER

Thème :

*Compression d'images numériques avec et
sans perte*

Présenté par :

- ❖ Benhizia Soundous.K
- ❖ Aroub Inas

Encadré par :

Khaldi Amine

2021/2022

Résumé

Le développement remarquable dans le domaine des technologies de l'information et la diversité des applications multimédias dans les dernières années, implique le développement des techniques de compression d'image efficaces, dont l'objectif est d'améliorer la capacité de transmission et le stockage des données.

Le but de ce travail est de réaliser une application de compression d'images numériques, deux approches seront proposées, une approche sans perte et une approche de compression avec perte. Les performances obtenues de ces deux approches seront comparées pour dégager la meilleure solution pour compresser les images numériques.

Mots-clés : Compression, Images numérique, Compression d'images, PSNR, SSIM, JPEG, JPEG 2000.

Abstract

The remarkable development in the field of information technology and the diversity of multimedia applications in recent years, involves the development of efficient image compression techniques, the objective of which is to improve the transmission and storage capacity data.

The purpose of this work is to achieve a digital image compression application, two approaches will be proposed, a lossless approach and a lossy compression approach. The performance obtained from these two approaches will be compared to identify the best solution for compressing digital images.

Index Terms : Compression, Digital Images, Image Compression, PSNR, SSIM, JPEG, JPEG 2000.

ملخص

التطور الملحوظ في مجال تكنولوجيا المعلومات وتنوع تطبيقات الوسائط المتعددة في السنوات الأخيرة، ينطوي على تطوير تقنيات فعالة لضغط الصور، الهدف منه هو تحسين بيانات سعة النقل والتخزين.

الغرض من هذا العمل هو تحقيق تطبيق ضغط الصور الرقمي، وسيتم اقتراح طريقتين، ضغط بدون فقدان وضغط مع فقدان. ستتم مقارنة الأداء الذي تم الحصول عليه من هاتين الطريقتين لتحديد أفضل حل لضغط الصور الرقمية.

كلمات المفاتيح : الضغط, الصورة الرقمية, ضغط الصور, JPEG 2000, SSIM, PSNR, JPEG.

Remerciements

Tout d'abord, nous voudrions remercier le Dieu Tout-Puissant et Miséricordieux de nous avoir donné de la patience dans notre travail.

Nous remercions sincèrement Monsieur KHALDI AMINE pour les conseils qu'il nous a prodigués, les connaissances qu'il nous a transmises, son suivi attentif et sa confiance inestimable en nous.

Nous remercions aussi sincèrement ceux qui nous ont aidés et contribué à l'élaboration de ce travail et à la réussite de cette belle année scolaire.

Nous remercions également tous les membres du jury d'avoir accepté de participer à l'évaluation de notre travail.

Nous tenons également à exprimer à toute l'équipe pédagogique et de direction du Master Académique d'Informatique, Spécialisation : Administration et Sécurité des Réseaux.

Enfin, un grand merci à toute la famille et les amis qui nous ont soutenus et encouragés tout au long du processus de thèse.

Dédicace

Soundous

Je dédie ce travail à :

À ceux qui sont les plus chers au monde, ma mère et mon père, à qui je n'arriverai jamais à exprimer ma gratitude, pour leur amour et leur soutien tout au long de mes études.

Ma mère a travaillé si dur pour ma réussite, pour son amour, pour son soutien, pour tous ses sacrifices et pour ses précieux conseils, pour toute son aide et sa présence dans ma vie, à travers ce travail, aussi dur soit-il, Soyez humble et exprimer mes sentiments et mon éternelle gratitude.

Mon papa peut fièrement retrouver ici le fruit d'années de sacrifices et de privations pour m'aider à avancer dans la vie. Que Dieu fasse fructifier ce travail ; merci pour vos nobles valeurs, votre éducation et votre soutien continu.

Mes frères, Mes sœurs.

Inas

J'ai l'honneur de dédie ce modeste travail à mes chers parents pour leur soutien moral et leurs sacrifices le long de ma formation, je ne pourrai jamais oublier d'exprimer ma gratitude à :

Mes sœurs (Abrar, Arwa et Alaa) et mes chers frères.

Je dédie ce travail à mes amies

A tous ceux qui me sont chères

A tous ceux qui m'aiment

A tous ceux que j'aime.

A Tous mes professeurs

Je dédie ce travail.

TABLE DES MATIERES

Résumé	
Remerciements	
Dédicaces	
Table des matières	
Liste des figures	
Liste des tableaux	
Liste des abréviations	
Introduction générale	13
CHAPITRE I : Notion sur les images numériques	14
I .1) Introduction	15
I .2) Définition d'une image	15
I .3) Image numérique	15
I .4) Les attributs de l'image	16
I .4) a. La Dimension	16
I .4) b. Le Bruit	16
I .4) c. Le Pixel	16
I .4) d. La Résolution	17
I .4) e. L'Histogramme	17
I .4) f. La Luminance	18
I .4) g. Le Poids de l'Image	18
I .4) h. Les Contours et Textures	18
I .4) i. Le Contraste	19
I .5) Nature des images	19
I .6) Les formats d'images	19
I .7) Les différents types d'images	22
I .7) a. L'image monochrome (binaire)	22
I .7) b. Image en niveaux de gris	23
I .7) c. L'image en couleurs	24
I .8) Etape de l'analyse d'une image	24

I .9) Domaines d'applications de traitement d'images	25
I .10) Conclusion	26
CHAPITRE II : Généralités sur les techniques de compression d'images	27
II .1) Introduction.....	28
II .2) Définition de la compression d'image	28
II .3) Schéma fonctionnel de la compression des images	28
II .3) a. Décorrélacion (Transformation)	29
II .3) b. Quantification	29
II .3) c. Codage	30
II .4) Classification des méthodes de compression	30
II .4) 1. Méthodes sans distorsion des données	30
4) 1.1 Codage de Shannon-Fano	31
4) 1.2 Codage de Huffman	32
4) 1.3 Codage arithmétique	33
4) 1.4 RLE ((Run Length Encoding)	34
II .4) 2. Méthodes avec distorsion des données	36
4) 2.1 Codage par quantification	36
4) 2.1.1 Quantification scalaire(QS)	37
4) 2.1.2 Quantification Vectorielle (QV)	38
4) 2.2 Méthodes par transformée	39
4) 2.2.1 Transformée en cosinus discrète (DCT)	40
4) 2.2.2 Transformée en ondelettes discrètes (DWT)	41
4) 2.3 Méthodes prédictives	42
4) 2.4 Méthodes hybrides	42
II .5) Les normes de compression des images avec et sans pertes	43
II .5) 1. La norme de compression JPEG	43
II .5) 2. La norme de compression JBEG2000	46
II .5) 3. La norme de compression JBIG	47
II .6) Codages imbriqués	47
II .6) 1. Algorithme EZW de Shapiro	47
II .6) 2. Algorithme SPIHT	48

II .6) 3. Algorithme SPECK	49
II .6) 4. Algorithme EZBC	49
II .7) Conclusion	49
CHAPITRE III : Résultats expérimentaux	51
III .1) Introduction	52
III .2) L'environnement de développement	52
III .3) Technique de compression	52
III .4) Tests et Résultats de la compression	54
III .4) 1. Tests sur l'image standard	54
III .4) 2. Ensemble de données et ses caractéristiques	55
III .4) 3. Formats d'image utilisés	55
II .4) 4. Les résultats et interprétations	60
4) 4.1 Compression d'image sans perte	60
4) 4.2 Compression d'image avec perte	62
III .5) Évaluation et Objectif	64
III .6) Conclusion	65
Conclusion Générale	66
Bibliographie	

LISTE DES FIGURES

CHAPITRE I : Notion sur les images numériques

Figure I.1 Un point de l'image de coordonnées(x, y).....	15
Figure I.2 Image numérique.....	16
Figure I.3 Représentation d'une image de groupe de pixels	17
Figure I.4 (A) Image de Lena (B) histogramme de l'image de Lena	18
Figure I.5 Contours et textures	19
Figure I.6 Image TIFF	20
Figure I.7 Image JPEG	21
Figure I.8 Une image binaire.....	23
Figure I.9 Image en niveaux de gris	23
Figure I.10 Image en couleurs.....	24

CHAPITRE II : Généralités sur les techniques de compression d'images

Figure II.1 : Schéma général de compression des images.....	29
Figure II.2 Codage de Shannon-Fano.....	32
Figure II.3 (a) et (b) Le codage de Huffman	33
Figure II.4 Principe de la quantification.....	37
Figure II.5 Quantification scalaire uniforme	37
Figure II.6 Principe du Quantificateur vectoriel	39
Figure II.7 Schéma de Compression –Décompression par transformée.....	40
Figure II.8 Application de la DCT sur chaque bloc.....	41
Figure II.9 Principe de l'algorithme JPEG avec pertes	43
Figure II.10 Principe de l'algorithme JPEG sans pertes	44
Figure II.11 Division de l'image en sous image 8*8	44
Figure II.12 Exemples de descendances parent-fils dans le cas SPIHT, le pixel désigné par (*) n'accepte pas de descendants.....	49

CHAPITRE III : Compression des images : Algorithmes, Simulation et Résultats

Figure III.1 : Icône MATLAB	52
Figure III.2 : L'organigramme de la compression d'une image	52
Figure III.3 : Images testes utilisées	56

Figure III.4 : L'interface générale de l'application.....	57
Figure III.5 : L'interface de l'image à compresser avec perte	57
Figure III.6 : Interface de résultats de compression avec perte.....	58
Figure III.7 : L'interface de l'image à compresser sans perte.....	58
Figure III.8 : Interface de résultats de compression sans perte	59
Figure III.9 : L'interface pour supprimer l'image d'origine.....	59
Figure III.10 : Interface de sortie	60
Figure III.11 : Image originale X	61
Figure III.12 : Image compressé Y	61
Figure III.13 : Image décompressé Z	61
Figure III.14 : Image originale A	61
Figure III.15 : Image compressé P	61
Figure III.16 : Image décompressé C	61
Figure III.17 : Image originale D	61
Figure III.18 : Image compressé E	61
Figure III.19 : Image décompressé F	61
Figure III.20 : Image originale H	62
Figure III.21 : Image compressé I	62
Figure III.22 : Image décompressé K	62
Figure III.23 : Image originale S	62
Figure III.24 : Image compressé F	62
Figure III.25 : Image décompressé B	62
Figure III.26 : Image originale N	63
Figure III.27 : Image compressé M	63
Figure III.28 : Image décompressé W	63
Figure III.29 : Image originale Q	63
Figure III.30 : Image compressé T	63
Figure III.31 : Image décompressé U	63
Figure III.32 : Image originale L	63
Figure III.33 : Image compressé J	63
Figure III.34 : Image décompressé V	63

LISTE DES TABLEAUX

CHAPITRE I : Notion sur les images numériques

Tableau I.1 : Tableau comparatif	22
---	----

CHAPITRE II : Généralités sur les techniques de compression d'images

Tableau II.1 : Fréquences (ou probabilité) des symboles	31
--	----

Tableau II.2 : Résumé des avantages et inconvénients de divers algorithmes de compression sans perte	36
---	----

Tableau II.3 : Les avantages et Les inconvénients de DCT	41
---	----

CHAPITRE III : Résultats expérimentaux

Tableau III.1 : Résultats des PSNR et SSIM (Compression sans perte)	62
--	----

Tableau III.2 : Résultats des PSNR et SSIM (Compression avec perte)	64
--	----

Tableau III.3 : Résultats des PSNR et SSIM	64
---	----

Liste des abréviations

PPP : Points par pouce

BMP: Windows Bitmap

TIFF: Tague Image File Format

GIF : Graphique Inter change Format

JPEG : Joint Photographique Experts Group

PNG : Portable Network Graphique

DCT: Discrète cosinus transform

DWT: Discret Wavelet Transform

EZW: Embedded zero tree wavelets

SPHIT: Set partitioning In Hierarchical trees

RLE: Run-length encoding

QS: Quantification Scalaire

QV: Quantification Vectorielle

JPEG: Joint Photographic Experts Group

MPEG: Moving Picture Exepert Group

PSNR: Peak Signal to Noise Ratio

SSIM: Structural Similarity Index

EQM : Erreur quadratique moyenne

INTRODUCTION GENERALE

La société actuelle génère de plus en plus de données, principalement des sons, des images ou des textes, issus de différents domaines tels que la physique, la médecine, la biologie, la culture, le tourisme ou la finance qui doivent être traitées, transmises et/ou stockées.

Cependant, ces techniques de compression sans perte conduisent à des taux de compression relativement faibles et sont dédiées à des applications sensibles, telles que la compression d'images médicales et satellitaires, et la compression de fichiers informatiques. L'accès aux grands appareils public et aux plates-formes hautes performances pousse les fournisseurs de contenu à offrir de plus en plus de choix dans la manière dont les images numériques sont diffusées. La compression implique de trouver des moyens de décrire les informations de manière aussi concise que possible, vous permettant éventuellement de les dégrader vous-même.

La compression d'une image peut être réalisée par projection basée sur une fonction orthogonale, puis transformée en codant le résultat de la transformation. Enfin, du fait de la grande quantité de données à traiter, il y a souvent un arbitrage entre la complexité des opérations réalisées et la qualité des résultats obtenus. Le principe général de toute compression est de réduire la taille des données en construisant des messages qui résumeront les informations initiales en éliminant la redondance.

Depuis le début de l'ère numérique, la quantité d'informations échangées sur les réseaux à travers le monde nécessite un traitement de l'information pour réduire la taille des données tout en préservant leur intégrité. Effets visuels de la dégradation de l'image, simplement parce que la numérisation de l'image réduit la qualité de l'image. Il comprend des techniques dites irréversibles pour obtenir des images reconstruites de qualité moyenne à des taux de compression élevés. Dans le cas particulier des images, la compression tente d'extraire des informations visuelles des données brutes reçues de la caméra. Cependant, la numérisation présente un inconvénient : elle nécessite une capacité de stockage importante et une bande passante de ligne de transmission suffisamment importante.

Ce mémoire est divisé en trois chapitres selon l'organisation suivante :

Le premier chapitre : nous allons étudier les différentes caractéristiques de l'image, les différents formats, quelques notions de base sur les images et enfin les traitements qu'on peut appliquer sur celles-ci.

Le deuxième chapitre : on a donné quelques définitions de la compression, tout en passant par les deux familles d'algorithmes : réversibles (sans perte) et irréversible (avec perte).

Le troisième chapitre : contient les résultats obtenus après l'application avec quelque discussion d'évaluation de notre modèle. Nous terminerons par une conclusion générale.

CHAPITRE I

Notions sur les images numériques

I. 1) Introduction :

L'imagerie a toujours été l'un des moyens les plus privilégiés que les humains utilisent pour communiquer avec les autres. En fait, c'est un moyen de communication universel dont le contenu riche permet à des personnes d'âges et de cultures différents de se comprendre. Chacun peut analyser les images à sa manière pour gagner des impressions et extraire des informations précises. Dans ce chapitre, nous allons introduire le concept d'images numériques de manière générale.

I. 2) Définition d'une image :

Elle peut être décrite sous la forme d'une fonction $I(x, y)$ de brillance analogique continue, définie dans un domaine borné, tel que x et y sont les coordonnées spatiales d'un point de l'image et en image numérique, cela compare au nombre de pixels qui créent l'image en hauteur (hachette verticale).

La définition d'une image est défini standard le nombre de se concentrer le composant. Le nombre de pixels qui la composent en largeur et en hauteur. Logiquement, in addition to une image comporte de pixels, in addition to elle est détaillée. La définition d'une image compare à son aspect, exprimée en pixels. : 200 pixels standards 450 pixels standards exemple, abrégé en « 200 × 450 ». Il est concevable de distinguer chaque pixel en grossissant fortement l'image. (1) (5) (8)

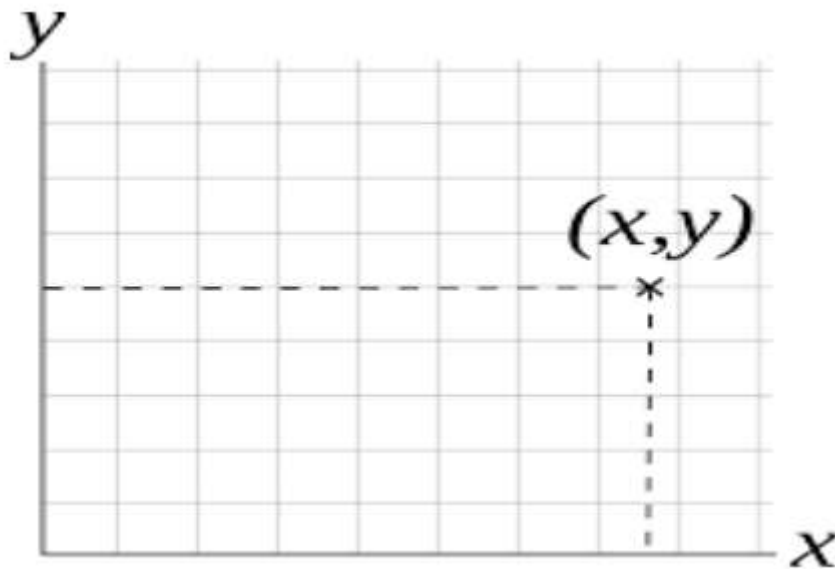


Figure I.1 Un point de l'image de coordonnées (x, y)

I. 3) Image numérique :

Une image numérique est constituée d'un ensemble de points appelés pixels. Un pixel (abréviation de Picture Élément) est défini comme le plus petit composant d'une image numérique raster. Pour les images bicolores en noir et blanc, les pixels peuvent être codés

par un seul bit de code (0 pour le noir ou 1 pour le blanc). Pour les images en niveaux de gris ou en couleur, les pixels peuvent être encodés en 2, 4, 8, 16, 24 ou 32 bits. (1) (8)

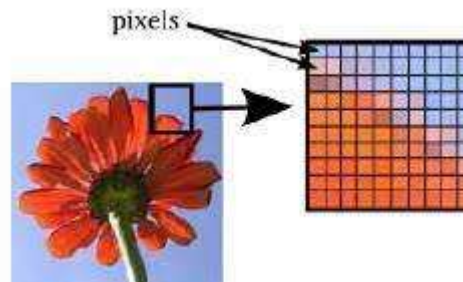


Figure I.2 Image numérique

I. 4) Les attributs de l'image :

L'image est un rassemblement structuré d'informations caractérisées standard le paramètre suivant : (1)

I. 4) a. La Dimension :

C'est la taille de l'image. Cette dernière se présente sous la forme d'une matrice dont les éléments sont des valeurs numériques représentant des intensités lumineuses (pixels). Le nombre de lignes de cette matrice multiplié par le nombre de colonnes nous donne le nombre total de pixels dans l'image. (1) (8)

I. 4) b. Le Bruit :

C'est un signal qui lors de l'acquisition ou la transmission vient s'ajouter à l'image.

Il se matérialise standard la présence dans une région homogène des valeurs in addition to ou moins éloignées de l'intensité de la région. Le bruit est le résultat de certains défauts électroniques du capteur et de la qualité de numérisation. (1) (8)

I. 4) c. Le Pixel :

À côté le nombre de pixels standard unité de longueur de la structure à numériser est élevé, in addition to la quantité d'information qui décrit cette structure est importante et in addition to la résolution est élevée. La résolution d'une image est définie standard un nombre de pixels standard unité de longueur de la structure à numériser (classiquement en ppp). Ce paramètre est défini lors de la numérisation (section de l'image sous forme binaire), et dépend principalement des caractéristiques du matériel utilisé lors de la numérisation. Ce n'est d'ailleurs pas la seule : la date et l'heure de prise de vue, entre autres, sont aussi des données stockées avec l'image. (1) (5) (8)

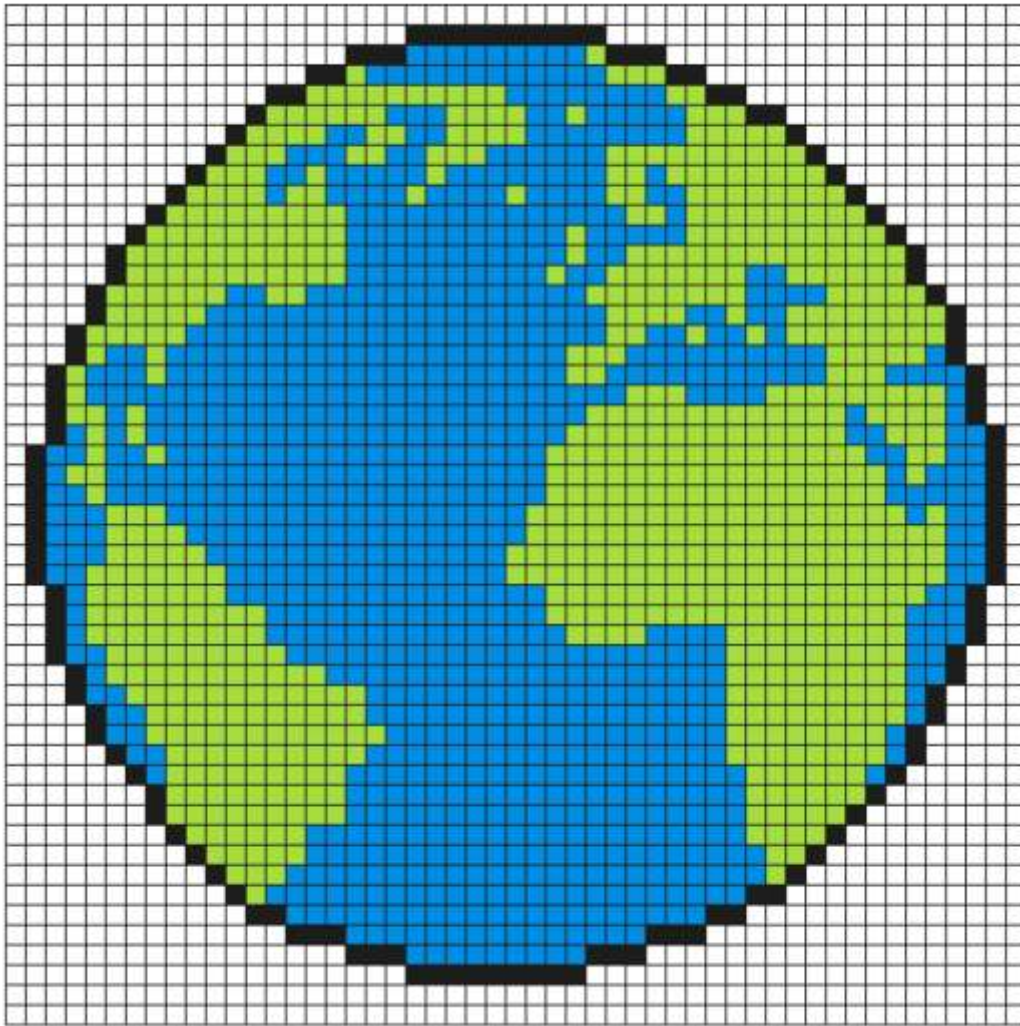


Figure I.3 Représentation d'une image de groupe de pixels

I. 4) d. La Résolution :

Est le nombre de pixels par pouce (1 pouce = 2,54 cm). Elle est exprimée en "DPI" (dots per inch) ou PPP (points par pouce). Plus il y a de pixels (ou de points) par pouce, plus il y a d'informations (et plus précises) dans l'image. (1) (8)

I. 4) e. L'Histogramme :

L'histogramme est une observation statistique, mais il est parfois utilisé comme modèle probabiliste empirique. L'histogramme des niveaux de gris ou des couleurs d'une image est une fonction qui donne la fréquence d'application de chaque niveau de gris (couleur) dans l'image. Pour diminuer l'erreur d'évaluation, pour comparer deux images obtenues sous des éclairages différents ou reprise pour mesurer les propriétés sur une image, on modifie souvent l'histogramme correspondant. (1) (8)

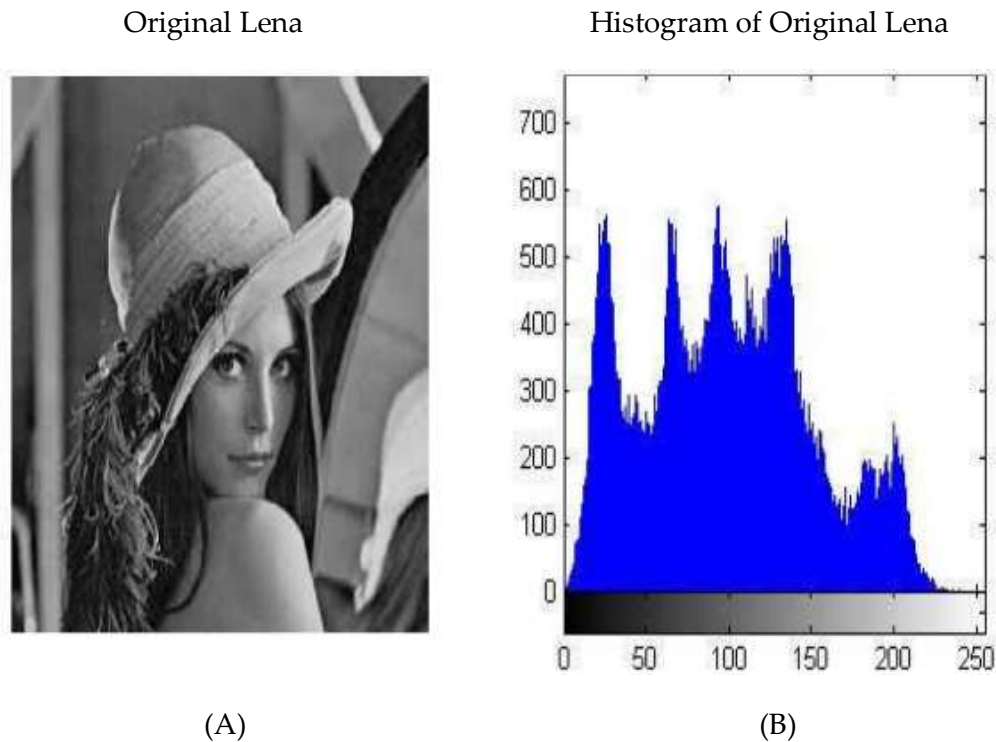


Figure I.4 (A) Image de Lena (B) histogramme de l'image de Lena

I. 4) f. La Luminance :

C'est le degré de luminosité de l'image sur laquelle vous vous concentrez. Elle est définie aussi comme le reste de l'intensité lumineuse d'une surface standard l'aire apparente de cette surface. (1) (8)

Pour un observateur lointain, le quolibet luminance est substitué à l'adage brillance, qui relate à l'éclat d'un objet. Une bonne luminance se caractérise standard :

1. Des images lumineuses (brillantes).
2. Un bon contraste.
3. L'absence de parasites.

I. 4) g. Le Poids de l'Image :

Le poids d'une image se détermine en fonction de trois paramètres : la dimension, la résolution et le nombre de couleurs. Il se mesure en octets (et ses multiples : ko, Mo, Go). (1)

I. 4) h. Les Contours et Textures :

Les contours représentent la frontière entre les objets de l'image, ou la limite entre deux pixels dont les niveaux de gris représentent une différence significative. Les textures décrivent la structure de ceux-ci. L'extraction de contour consiste à identifier dans l'image les points qui séparent deux textures différentes. (1) (8)



Figure I.5 Contours et textures

I. 4) i. Le Contraste :

C'est l'opposition marquée entre deux régions d'une image, plus précisément entre les régions sombres et les régions claires de cette image. (1) (8)

I. 5) Nature des images :

Les images à compresser peuvent être de différentes natures : (1)

✚ Les images (modales)

Dans lesquelles l'intensité lumineuse est très changeante localement (dessin manuels par exemple). Leur histogramme est multimodal.

La numérisation des images peut aussi se faire de différentes manières, retrouvant :

- **les images binaires** : qui contiennent seulement deux niveaux de gris différents.
- **les images multi-niveaux** : numérisées sur plus d'un (1) bit par pixel.
- **les images multicanaux** : dont l'exemple classique est celui des images couleurs.
 - ✚ Image photographiques
- Ce sont généralement des images de scènes naturelles dans lesquelles l'intensité lumineuse varie de manière relativement continue.

I. 6) Les formats d'images :

Un design d'image est une représentation informatique de l'image, associée à des informations sur la façon dont l'image est codée et fournissant éventuellement des signes sur la manière de la décoder et de la manipuler. (1)

On distingue plusieurs types de formats d'images à savoir les images :

➤ **BMP (Windows Bitmap) :**

Diverses profondeurs de couleur, canaux alpha, profils de couleur et pression de données optionnelles sont supportés dans ce design. Le design d'image BMP est largement utilisé sur la plate-forme Windows et ce design est totalement viable avec PaintShop Pro. (1) (14) (13)

CHAPITRE I : Notion sur les images numériques

Ce design est l'un des premiers qui a été utilisé en informatique.

Le design BMP est un des premiers designs d'image utilisée sous Windows. Il est un des seuls designs à ne pas utiliser à pressurer. Le fichier BMP contient des données graphiques matricielles qui sont indépendantes des périphériques d'affichage. Les images BMP sont généralement non compressées ou compressées avec une méthode de pressuré sans perte.

➤ TIFF (Tage Image File Format) :

En dépit de leur grande consommation en matière d'espace de stockage, ils sont in addition To fiable pour ce qu'il en ait des impressions de haute résolution dans le domaine du matériel. Il permet d'obtenir une image de très bonne qualité, mais sa taille reste volumineuse, même si elle est inférieure à celle des fichiers BMP, véhicule les fichiers TIFF peuvent être compressés largement pris en charge standard les différentes plateformes existantes, il est souvent approprié dans les environnements professionnels et précisément dans le domaine business. (1) (5) (14) (13)

Ce design est orienté vers les professionnels (imprimeurs, publicitaires...) véhicules, il a l'avantage d'être reconnu sur tout type de système d'exploitation : Windows, Mac, Linux, Unix...



Figure I.6 Image TIFF

➤ GIF (Graphique Inter change Format) :

Les fichiers GIF sont de petite taille, ce qui est dû au courant que ces images ne peuvent enregistrer que 256 couleurs.

Le GIF est un design qui offre une image compressée promote en gardant bon rendu, mais qui reste limité au niveau de la profondeur de couleur (limité à 256 couleurs). Ils sont donc pris en charge standard la plupart des navigateurs utilisant des algorithmes de pressurer pour garder leurs tailles initiales. Contrairement au JPG, ils autorisent les fonds transparents et peuvent aussi stocker plusieurs images afin de créer des diapos ou des liveliness.

CHAPITRE I : Notion sur les images numériques

Les GIF texte le style part des fichiers multimédia les in additions to utilisés sur le web et pour les messages électroniques. (1) (5) (14)

➤ **JPEG (Joint Photographique Experts Group) :**

Ce design offre des taux de pressurer inégalés, même si la qualité de l'image s'en ressent au fur et à mesure que vous augmentez la pression.

Avec des taux de pressuré élevés donnants lieux à des fichiers imagent de modeste taille, ce design est devenu le standard des designs d'image sur le web. Généralement, l'on opte pour une résolution basse pour les messages, une résolution moyenne pour les designs web et une grande résolution pour les archives commerciales.

Les JPG sont une expansion des design JPEG qui est une norme de pressuré d'image numérique. (1) (5) (14) (13)



Figure I.7 Image JPEG

➤ **PNG (Portable Network Graphique) :**

Comme le GIF, il permet le détournage des images, mais là où le design GIF enregistre 256 couleurs, le PNG en retient 16.7 MILLIONS ce qui offre une image parfaite, avec un fantastique rendu des subtilités et des dégradés.

Les PNG sont des designs de fichiers créés pour succéder au design GIF vu qu'elles offrent un meilleur résultat de pressuré sans perte de qualité (entre 5% et 25% de qualité conservée).

La taille des fichiers reste raisonnable, et, technologie dont ce design est le seul à disposer, il permet la pression sans perte de donnée ! En effet, les PNG sont recommandés standard les W3C pour l'affichage des icônes, des logos, des images représentant des textes.

C'est le futur standard pour web, véhicule, il autorise la transparence comme le design GIF et autorise 16.7 millions de couleurs. Ce sont des images bitmap qui utilisent un algorithme de pressuré sans perte de données. (1) (5) (14) (13)

CHAPITRE I : Notion sur les images numériques

➤ FAX :

Ce design utilisé pour la transmission de reports (télécopie), codage binaire, application d'un codage RLC puis d'un codage de Huffman. (1)

➤ TGA :

Le design TGA a été créé à l'origine standard la firme True Vision, fabriquant des cartes graphiques Targa et Vista. Ce design est parmi les designs les plus puissants, avec le design TIFF. Toutefois, il est moins connu que les autres designs comme le design BMP, le design EPS. Il peut coder la couleur sur 8, 16, 24 ou 32 bits par pixel standard. Ce design est aussi compétent de gérer le noir et blanc et le RGB. (1)

📊 Tableau comparatif : (5)

	Type (matriciel/vectoriel)	Compression des données	Nombre de couleurs supportées	Affichage progressif	Animation	Transparence
JPEG	matriciel	Oui, réglable (avec perte)	16 millions	Oui	Non	Non
GIF	matriciel	Oui, Sans perte	256 maxi (palette)	Oui	Oui	Oui
TIFF	matriciel	Compression ou pas avec ou sans pertes	de monochrome à 16 millions	Non	Non	Oui (couche Alpha)
PNG	matriciel	Oui, sans perte	Palettisé (256 couleurs ou moins) ou 16 millions	Oui	Non	Oui (couche Alpha)

Tableau I.1 Tableau comparatif

I. 7) Les différents types d'images :

Il existe différentes catégories d'images: (1)

I. 7) a. L'image monochrome (binaire) :

Les images binaires sont des images en niveau de gris particulières, dont les valeurs valent soit 0 (noir), soit 1 (blanc). On cherche souvent à se ramener les images de départ à des images binaires, puisque il est très facile ensuite d'identifier les objets présents. L'image

obtenue n'est pas très nuancée, alors, pour convertir une image couleur en mode monochrome, il faut d'abord passer standard le mode niveaux de gris. (1) (5) (13)



Figure I.8 Une image binaire

I. 7) b. Image en niveaux de gris :

La plupart des images designs de fichiers prennent en charge un least de niveaux de gris en bits 8, qui fournissent les niveaux de luminance 2^8 ou 256 standards pixels. Les images peuvent être enregistrées en niveaux de gris (ou en noir et blanc), même les images couleur contiennent des informations en niveaux de gris.

De nombreux programmes d'édition d'images vous permettent de convertir une image couleur en noir et blanc ou en niveaux de gris. Certains programmes fournissent d'autres changes de niveaux de gris personnalisé algorithmes qui vous permettent de générer une image en noir et blanc avec l'apparence que vous préférez. Les images numériques étant affichées à l'aide d'une combinaison de rouge, de vert et de bleu (RGB) couleurs, chaque pixel a trois valeurs de luminance distinctes. Certains designs prennent en charge les niveaux de gris en bits 16, qui fournissent les niveaux de luminance 2^{16} ou 65,536. Standard conséquent, une image en niveaux de gris ne contient que des subtilités de gris et aucune couleur. Un choix consiste à faire la moyenne de toutes les valeurs de luminances pour chaque pixel. (1) (5) (8) (13)



Figure I.9 Image en niveaux de gris

I. 7) c. L'image en couleurs :

On peut représenter les couleurs à l'aide de leurs composants primaires, le rouge, le vert et le bleu (le modèle R.V.B). Les images en couleurs vraies stockent pour chaque pixel 3 composantes : rouge, vert, bleu (ou RGB, pour Red-Green-Blue). L'espace HSV décompose chaque pixel en trois composantes de teinte (tint, qui représente la couleur principale), d'immersion (dose entre « coloré » et « gris »), et de valeur (qui correspond à l'intensité globale du pixel). La représentation des couleurs s'effectue de même manière que les images en niveaux de gris avec cependant quelque particularité. Ces trois composantes permettent de représenter toutes les autres couleurs, y compris les subtilités de gris. Si chaque composante est codée sur 8 pièces (soit valeurs 256 possibles), on peut ainsi représenter jusqu'à $256*256*256=16$ millions de couleurs différentes. (1) (5) (8) (13)



Figure I.10 Image en couleurs

Il y a des différentes représentations des images couleurs :

- La représentation en couleurs réelle sur 24 bits.
- La représentation en couleurs indexée ; on utilise une table appelée palette pour éviter la redondance de couleurs.
- Une image RVB est composée de trois couches: Rouge, Vert et Bleu. Chaque pixel est défini par une valeur possible de ces couleurs de [0 à 255]. Une fois combinées, ces couches permettent de générer toutes nuances de couleurs.

I. 8) Etape de l'analyse d'une image :

Les étapes de traitement d'images sont: (1)

- L'acquisition, l'échantillonnage, la quantification.
- L'analyse globale de l'image et transformations ponctuelles : Histogramme ; statistiques (moyenne, écart-type ...), transcodage (palette de couleur [LTU]) et classification.
- L'acquisition, l'échantillonnage, la quantification.

I. 9) Domaines d'applications de traitement d'images :

Cette data peut prendre les trois formes suivantes : (1)

Une image, standard exemple pour l'amélioration de la représentation : on parlera alors de traitement de Bas Niveau, de valeurs numériques, éventuellement associées à une image, standard exemple l'extraction de sections de formes, les valeurs numériques étant les coordonnées des se concentre extrémités. On parlera de traitement de Niveau Intermédiaire, de valeurs symboliques, ou de décisions : standard exemple :

CHAPITRE I : Notion sur les images numériques

Un cercle, une droite, ou un objet prédéfini ont été reconnus dans l'image, une pièce manufacturée a été reconnue conforme : le préhenseur du bras de robot, ou du robot delta va prendre la pièce sur le tapis roulant et la stocker, un singe a été reconnu : le tir va être effectué.

De nouvelles applications pratiques possibles aujourd'hui et qui touchent tous les domaines d'activités, tels que : les métiers du spectral, de la radio, les créations artistiques. On trouve ses applications dans des domaines très variés tels que les télécommunications (TV, vidéo, publicité...), la médecine (radiographie, ultrasons...).

Les exemples et domaines d'applications du traitement d'image sont très nombreux. Les deux principaux domaines, qui ont permis au Traitement d'Image de se développer sont : (12)

✚ Le domaine militaire :

- Le renseignement (télédétection à partir d'images satellite dont la précision peut aller aujourd'hui jusqu'à quelques centimètres ou aériennes, la photographie interprétation).
- Rockets en toutes sortes (autodirecteur (courte portée), de croisière (longue portée), etc).

✚ Le domaine médical :

- Angiographie
- Radiographie
- Echographie
- Scanner
- IRM

Mais, une multitude d'applications apparaissent dans des domaines divers allant :

✚ Du domaine industriel :

- Les contrôles automatiques standards des robots (« choisir et repérer » : récupération sur un tapis roulant pour mise en sachet ou dans un conteneur) y compris dans l'industrie alimentaire en in addition to du contrôle de la structure, de la fraîcheur.
- Le contrôle de la qualité des produits en session de chaîne (état de surface, dimensionnement, forme, couleur, présence des comprimés standard exemple dans l'industrie pharmaceutique).

✚ Allant jusqu'au domaine du multimédia :

- La compression / décompression pour la transmission d'Images ou le stockage
- L'amélioration de la qualité
- La TV haute définition (Standard MPEG4)

En passant par de nombreux domaines dont :

CHAPITRE I : Notion sur les images numériques

- ✚ L'imagerie civile satellitaire et aérienne
- ✚ La reconnaissance de caractères, de documents
- ✚ La robotique mobile et autonome militaire
- ✚ La régularisation de flux par le biais de comptage et suivi d'objets
- ✚ La surveillance de sites ou locaux stratégiques, protection de données

I. 10) Conclusion :

Dans ce chapitre, nous avons présenté les notions d'images d'une manière générale. Nous avons essayé de donner un aperçu sur les différents formats, caractéristiques et les différents types d'images. Cependant, nous allons dans le chapitre suivant présenter les différentes techniques de compression.

CHAPITRE II

Généralités sur les techniques de compression d'images

II. 1) Introduction :

Malgré l'augmentation de la capacité des supports de stockage et des débits des réseaux, il existe toujours un besoin de compresser les images compte tenu de l'évolution des usages de l'image et de la vidéo et de l'augmentation de la capacité d'acquisition des capteurs. Les méthodes de compression sans perte sont dédiées aux applications sensibles, telles que la compression d'images médicales et satellitaires, et la compression de fichiers informatiques, un engouement qui ne se dément pas depuis les années 1970, porté par l'évolution des méthodes d'acquisition, des capacités de stockage et des performances de calcul.

Dans ce chapitre, nous donnons quelques concepts de base sur la compression d'image, les paramètres qui permettent d'évaluer ses performances et les différentes méthodes de compression. La compression de cette grande quantité d'informations est donc une étape importante pour optimiser la taille mémoire nécessaire à son archivage et réduire le temps nécessaire à la télétransmission sur un réseau informatique. Les opérations de compression d'image sont utiles pour l'archivage ou le transfert sur support de stockage. Une légère perte (parfois imperceptible) de fidélité est acceptable pour les images naturelles (telles que les photographies) afin d'obtenir des réductions substantielles du débit binaire.

II. 2) Définition de la compression d'image :

Les méthodes de compression et d'encodage réduisent le nombre de bits par pixel à stocker ou à transmettre en exploitant la redondance des informations dans l'image.

La compression d'image est une application de compression de données sur des images numériques. Son utilité est de réduire la redondance des données dans une image afin qu'elle puisse être stockée ou transférée rapidement sans prendre beaucoup de place. La langue elle-même utilise des mots de longueurs variables, les plus courants étant les plus courts, pour réduire la taille des phrases.

Le compresseur utilise un algorithme qui optimise les données avec des considérations spécifiques au type de données compressées. Par conséquent, un décompresseur est nécessaire pour reconstruire les données d'origine en utilisant l'algorithme inverse utilisé pour la compression. (1) (9) (3)

Les principaux critères d'évaluation de toute méthode de compression sont :

- La qualité de reconstitution de l'image.
- Le taux de compression.
- La rapidité du codeur et décodeur (codec).

II. 3) Schéma fonctionnel de la compression des images :

Le schéma fonctionnel de la compression est présenté dans la figure II.1 ci-dessous : (1) (9) (3)

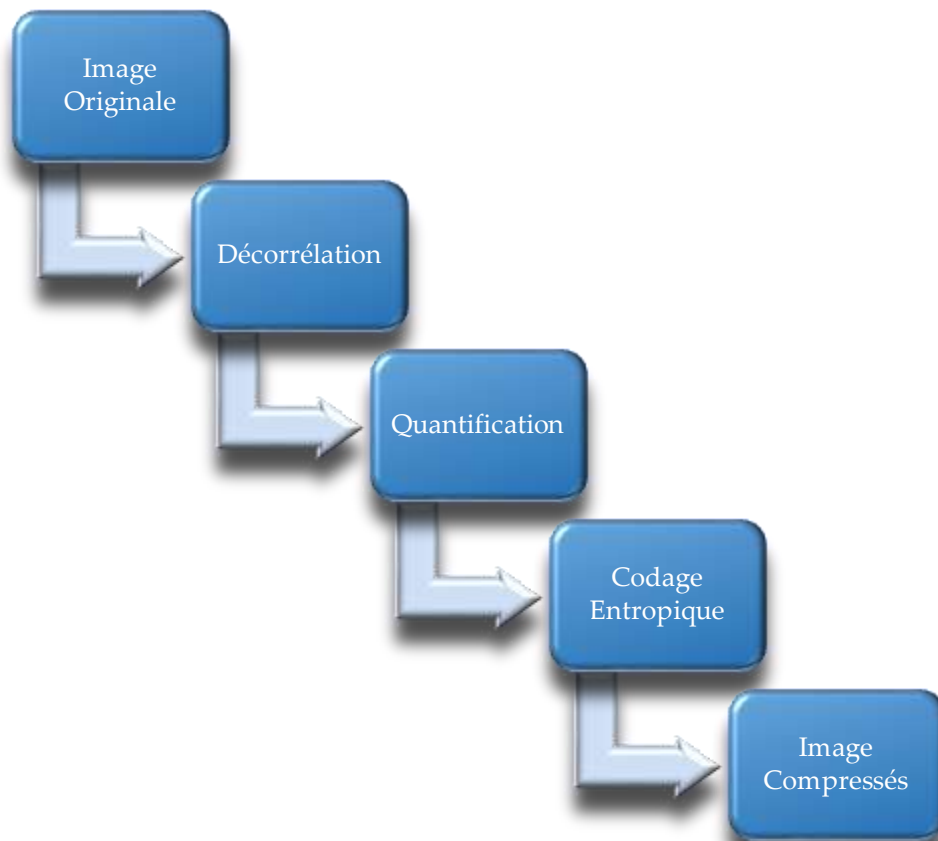


Figure II.1 : Schéma général de compression des images

A partir de ce schéma, nous allons revoir chacune de ses étapes à fin de préciser leur rôle.

II. 3) a. Décorrélation (Transformation) :

En effet, en plus de réorganiser l'information, il doit également représenter les composantes importantes du signal avec le moins d'éléments possible : c'est ce qu'on appelle donner une représentation parcimonieuse du signal, ou de manière équivalente, compresser l'énergie.

Le but d'une transformation est de projeter un signal sur la base d'une fonction dont les propriétés sont appropriées aux propriétés et caractéristiques du signal que l'on souhaite analyser.

Les pixels qui composent une image numérique sont souvent liés, ce qui crée des informations redondantes qui doivent être utilisées pour la compression afin de réduire la taille de l'image.

Les dépendances existantes entre chaque pixel et ses voisins (petits changements de luminosité d'un pixel aux voisins) reflètent de très fortes corrélations dans l'image. En effet, en plus de réorganiser l'information, il doit représenter les parties importantes du signal avec le moins d'éléments possible. (1) (3) (10)

II. 3) b. Quantification :

CHAPITRE II : Généralités sur les techniques de compression d'images

En réduisant la précision de la transformée, la quantification réduit le nombre de bits nécessaires pour stocker les coefficients transformés. L'un des résultats fondamentaux des travaux de Shannon sur la relation : (débit/distorsion) montre que de meilleures performances peuvent être obtenues en utilisant la quantification vectorielle.

La quantification des coefficients vise à réduire le nombre de bits nécessaires à leur représentation.

Dans les schémas de compression, l'étape de quantification est une étape qui dégrade de manière irréversible la qualité de l'image. Il se rapproche de chaque valeur du signal comme un multiple entier de la quantité q , appelée quantum fondamental ou pas de quantification. Il se rapproche de chaque valeur du signal comme un multiple entier de la quantité q , appelée quantum fondamental ou pas de quantification. (1) (3) (9) (4) (10)

II. 3) c. Codage :

Cette dernière étape est nécessaire dans les méthodes sans perte, mais elle se produit aussi souvent dans les algorithmes irréversibles, car les valeurs transformées et quantifiées contiennent plus de redondance. Le codeur doit répondre a priori aux deux conditions suivantes :

- Unicité : Deux messages différents ne peuvent pas être encodés de la même manière.
- Déchiffrement : Deux mots de code consécutifs doivent être distingués sans ambiguïté.

Cependant, l'absence d'un codeur entropique peut être raisonnable et nécessaire, notamment en raison des limitations de transmission.

Le codage entropique effectue un codage nul sur les valeurs quantifiées. (1) (3) (9) (4) (10)

II. **4) Classification des méthodes de compression :**

II. 4)1. Méthodes sans distorsion des données :

Cette technique de compression conservatrice est utilisée dans des applications telles que l'archivage d'images médicales, l'imagerie satellite, où les photographies sont coûteuses, et les informations critiques, textes, programmes et toute autre forme de données devant être stockées dans une résidence. Si les données sont compressées sans perte, les données d'origine peuvent être récupérées avec précision à partir des données compressées.

La compression d'image sans perte est un algorithme de compression qui reconstruit parfaitement l'image d'origine à partir des données d'origine.

Avec la compression sans perte, les données sont compressées sans perte de données. Comme leur nom l'indique, les techniques de compression sans perte n'impliquent pas de perte d'informations.

Ils peuvent trouver exactement les pixels de l'image numérique d'origine. (1) (3) (5) (10) (7) (8)

➤ 4)1.1 Codage de Shannon-Fano

Shannon du Bellis Lab et R.M. Fano du MIT ont développé une méthode de codage basée sur une simple connaissance de la probabilité d'occurrence de chaque symbole dans un message presque simultanément. Ils utilisent des codes de longueur variable, comprenant plus de bits, car la probabilité de symboles est faible.

Le processus de Shannon-Fano construit un arbre descendant à partir de la racine par partitions successives. Le tri des fréquences est effectué par ordre décroissant, en supposant que le fichier est lu en premier et que les en-têtes sont enregistrés. (9) (1) (3) (5) (7) (8)

Le principe est le suivant :

- Les symboles sont triés et classés en fonction de leur fréquence, commençant par le plus fréquent.
- La liste des symboles est ensuite divisée en deux parties de manière à ce que le total des fréquences de chaque partie soit aussi proche que possible.
- Le chiffre binaire 0 est affecté à la première partie de la liste, le chiffre 1 à la deuxième partie.
- Rediviser chaque sous ensemble en deux nouveaux sous-ensembles de fréquences équivalentes afin de réitérer l'algorithme jusqu'à ce qu'il n'y ait plus qu'un élément dans chaque sous ensemble.

Exemple :

La Chaîne que nous allons traiter est : ACBBCDECECEBCEEECCCCABCECECBDBDBD

La table des fréquences (ou probabilité) des symboles :

Symbole	Fréquences
C	11
B	7
E	5
D	4
A	2

Tableau II.1 Fréquences (ou probabilité) des symboles

Voici le développement du calcul sous forme de schéma :

CHAPITRE II : Généralités sur les techniques de compression d'images

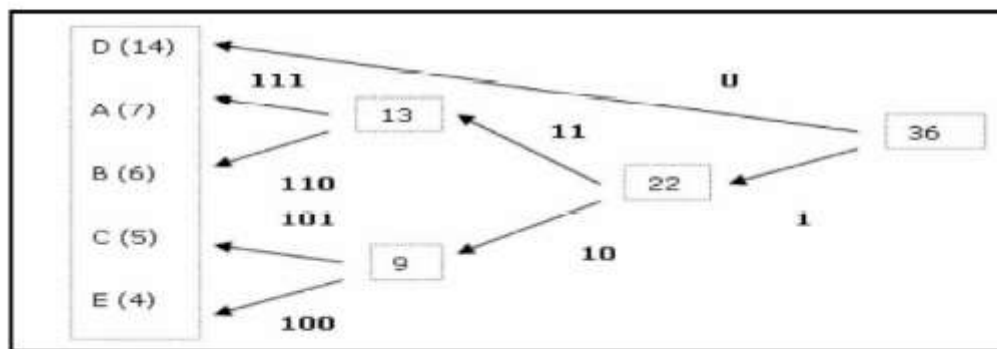
- Regrouper par séquences les paires de symboles de plus faible probabilité, en les reclassant si nécessaire. Plus précisément : calculer $s = f_i n + f_{i-1}$, la somme des deux plus faibles fréquences.
- Choisir le plus petit indice k tel que s soit supérieur ou égal à f_k , remplacer k par $k+1$.
- Recomposer la table des fréquences en plaçant à la k ème position la valeur s et en décalant les autres d'une position vers le bas. Puis décrémenter n d'une unité, poursuivre jusqu'à ce que la table des fréquences ne comporte plus que deux éléments.
- Coder avec retour arrière depuis le dernier groupe, en ajoutant un 0 ou un 1 pour différencier les symboles préalablement regroupés.

Exemple:

Soit le tableau suivant :

Symboles	D	A	B	C	E
Fréquences	14	7	6	5	4

(a)



(b)

Figure II.3 (a) et (b) Le codage de Huffman

Les codes obtenus par la méthode de Huffman sont: D(14)=0. A (7) =111. B (6) =110. C (5) =101. E (4) =100.

➤ 4)1.3 Codage Arithmétique

Par exemple, un symbole avec une probabilité de 0,9 a une entropie de 0,15, mais Huffman peut attribuer un code d'un bit (ou plus), et la séquence codée sera plus longue en bits que théorique. Ce code de sortie est un nombre à virgule flottante entre 0 et 1 avec le nombre de chiffres après la virgule correspondant au nombre de symboles. En fait, au lieu d'attribuer un mot de code à chaque symbole, il associe une valeur dans l'intervalle $[0,1]$ à un ensemble de symboles.

CHAPITRE II : Généralités sur les techniques de compression d'images

Contrairement aux algorithmes de codage qui utilisent des mots de code de longueur variable, ici les codes sont associés à des séquences plutôt qu'à chaque symbole acquis individuellement. Après lecture de chaque symbole d'entrée de l'alphabet, l'intervalle est subdivisé en intervalles plus petits proportionnels à la probabilité du symbole d'entrée.

Contrairement aux algorithmes de Huffman et de Shannon-Fano, ce dernier associe des symboles à des motifs binaires dont les grandeurs dépendent de leur distribution.

Le codage arithmétique se distingue par sa capacité à un ensemble de symboles de nombre non entier de bits.

Dans cette section, nous présentons les idées principales du codage arithmétique, l'une des techniques de compression sans perte les plus efficaces et probablement la plus importante dans les applications et les normes actuelles. Contrairement à ce dernier, il génère un code pour toute la séquence de symboles, plutôt qu'un code par symbole. Ce sous-intervalle devient alors le nouvel intervalle et est divisé en parties selon les probabilités des symboles alphabétiques entrés. (9) (1) (3) (5) (8)

Les encodeurs arithmétiques sont plus efficaces que les encodeurs Huffman, mais sont plus complexes à mettre en œuvre.

Enfin, tous les nombres à virgule flottante dans le dernier intervalle déterminent de manière unique les données d'entrée.

Le codage arithmétique est le codage le plus récent qui utilise des modèles statistiques, comme les encodeurs Huffman.

Algorithme du codage arithmétique :

Nous décrivons brièvement ci-dessous l'algorithme de codage arithmétique dans le but d'en illustrer le principe, sachant que le décodage opère de manière inverse.

- **Etape 1 :** Calculez la probabilité associée à chaque symbole de la chaîne à encoder.
- **Etape 2 :** Dans l'intervalle [0,1], est associé à chaque symbole un sous-intervalle proportionnel à sa probabilité (on retiendra l'ordre des intervalles, car il est requis par le décodeur).
- **Etape 3 :** Initialiser la limite inférieure de l'intervalle de travail à la valeur 0 et la limite supérieure à la valeur 1. Tant qu'il reste un symbole dans la chaîne à coder:
 - largeur = limite supérieure - limite inférieure.
 - limite inférieure = limite inférieure + largeur * (limite basse du sous intervalle du symbole).
 - limite supérieure = limite inférieure + largeur * (limite haute du sous intervalle du symbole).
- **Etape 4 :** La limite inférieure code la chaîne de manière unique.

➤ 4)1.4 RLE (Run Length Encoding)

CHAPITRE II : Généralités sur les techniques de compression d'images

Ce codage consiste, pour chaque séquence de pixels de même couleur, à ne coder que le nombre de pixels, puis la couleur de la séquence. Au lieu de coder des symboles consécutifs de la chaîne, le nombre de symboles consécutifs de même valeur est codé. Le principe est de regrouper une même valeur adjacente et de ne transmettre la valeur qu'une seule fois, précédée du nombre de répétitions.

L'utilisation de cette méthode a commencé avec l'avènement de la théorie de l'information.

Dans le cas de la compression d'images, RLE est utilisé par de nombreux formats d'images (BMP, PCX, TIFF). Cependant, sans répétition d'éléments, cette technique ne donne pas de résultats satisfaisants. Il existe des variantes où l'image est encodée par des blocs de points, le long de lignes ou même en zig-zag. Le plus simple de ces algorithmes est le codage de longueur de plage (RLE), une méthode également connue sous le nom de codage de longueur de plage (RLC).

Il existe plusieurs algorithmes utilisés pour compresser les informations des plus simples aux plus complexes et leur efficacité varie avec celle-ci. Évidemment, cette méthode fonctionne bien s'il y a beaucoup de répétitions dans le signal. (9) (1) (3) (5) (7) (8)

Avantages et inconvénients de la compression sans perte : (5)

Des techniques	Avantages	Désavantages
Codage de Huffman	<p>cet algorithme de compression très simple et efficace compressant des fichiers texte ou programme.</p> <p>Cette technique montre des séquences plus courtes pour des caractères plus fréquents.</p> <p>Sans préfixe: aucun codage de séquence de bits d'un caractère n'est le préfixe de tout autre codage de séquence de bits.</p>	<p>Une image compressée par cette technique est mieux compressée par d'autres algorithmes de compression.</p> <p>L'arbre de code doit également être transmis ainsi que le message (sauf si une table de code ou une table de prédiction est convenue entre l'expéditeur et le destinataire).</p> <p>Données entières corrompues par un bit corrompu.</p> <p>La performance dépend d'une bonne estimation si l'estimation n'est pas meilleure que la performance est mauvaise.</p>

<p>Codage Arithmétique</p>	<p>sa capacité à garder le codage et le modeler séparés.</p> <p>aucune arborescence de code ne doit être transmise au récepteur.</p> <p>son utilisation les valeurs fractionnaires.</p>	<p>le codage arithmétique a des opérations complexes car il consiste en additions soustractions multiplications et divisions.</p> <p>codage arithmétique nettement plus lent que le codage de Huffman, il n'existe aucune précision infinie.</p> <p>deux structures de problèmes pour stocker les nombres et la division constante de l'intervalle peuvent entraîner un chevauchement de code.</p>
-----------------------------------	--	---

Tableau II.2 Résumé des avantages et inconvénients de divers algorithmes de compression sans perte

II. 4.2 Méthodes avec distorsion des données

Le taux de compression d'un facteur 4, n'est pas toujours suffisant pour la transmission et le stockage des images numériques de grandes tailles. Pour l'améliorer, il va falloir perdre de l'information et par conséquent dégrader l'image. Il existe plusieurs méthodes pour la compression données :

- par quantification
- par prédiction
- par transformation

Ces méthodes permettent de retrouver une approximation de l'image numérique.

Les méthodes irréversibles permettent des taux de compression assez élevés au prix d'une dégradation de la qualité de l'image. Outre, le taux de compression, une mesure de cette dégradation est nécessaire à l'évaluation des performances de ces méthodes. Des algorithmes de compression avec pertes est de minimiser cette dégradation pour un taux de compression donné. (1) (3) (5) (10) (7) (8)

➤ 4)2.1 Codage par quantification

En réduisant la précision de la transformée, la quantification réduit le nombre de bits nécessaires pour stocker les coefficients transformés. Ces éléments peuvent être des scalaires, lorsque les informations de chaque pixel sont remplacées par des nombres entiers ; ou des vecteurs lorsque l'on considère des groupes de pixels représentant des configurations typiques.

Souvent, la quantification se produit comme un deuxième bit dans le processus de compression pour réduire la quantité d'informations, généralement de manière irréversible. Par conséquent, la valeur initiale est remplacée par un ensemble limité d'éléments, qui donneront des résultats acceptables lors de la phase de décompression. La quantification peut être appliquée à des coefficients individuels (quantification scalaire QS).

Par conséquent, la quantification est utilisée pour simplifier la représentation tout en préservant les informations. (9) (1)

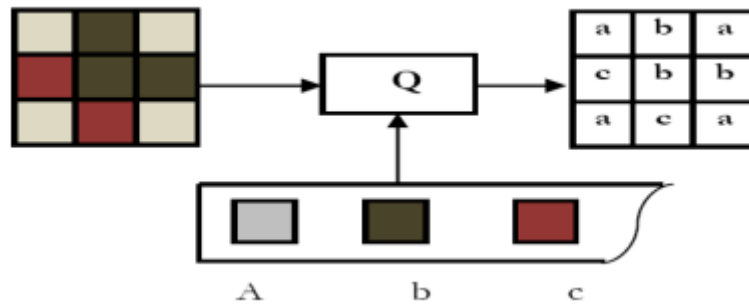


Figure II.4 Principe de la quantification

❖ 4)2.1.1 Quantification scalaire(QS)

Un quantificateur scalaire est un opérateur qui associe une variable continue u à une variable discrète u' pouvant prendre un nombre petit et fini de valeurs ; il est généralement défini comme une fonction en escalier (Figure II.5), avec chaque valeur L l'intervalle entre est appelé l'intervalle de décision.

Les quantificateurs scalaires sont des opérateurs qui relient des variables continues à des variables discrètes, qui peuvent prendre un nombre petit et fini de valeurs. Il s'agit d'une opération irréversible largement utilisée en compression (il est à noter que les procédés de codage utilisant des quantificateurs ne sont jamais réversibles, car l'étape de quantification introduit inévitablement de la distorsion). (9) (1) (8)

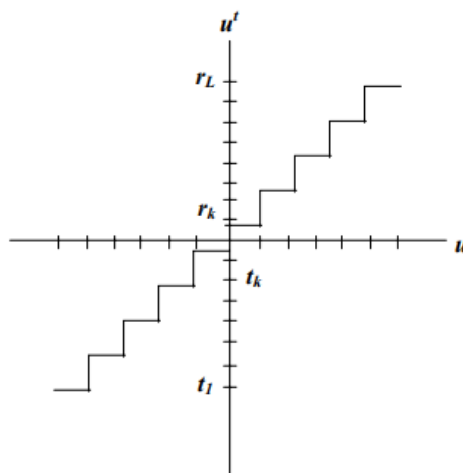


Figure II.5 Quantification scalaire uniforme

❖ 4)2.1.2 Quantification Vectorielle (QV)

QV, développé par Gersho et Gray, occupe une place très importante dans le domaine de la compression d'images, que ce soit à des fins de transmission ou d'archivage.

La quantification vectorielle, développée par Gersho et Gray (1980), fait aujourd'hui l'objet de nombreuses publications dans le domaine de la compression d'images numériques. Le rôle de l'encodeur est de rechercher dans le dictionnaire Y le code vectoriel y le plus proche pour tout vecteur x du signal d'entrée (patch). Il s'agit d'une quantification vectorielle incluant les mêmes opérations que QS, seul le partitionnement se fait dans R . (9) (1) (3) (8)

R : Un ensemble de nombres réels.

K : Dimension des vecteurs utilisés.

QV est divisé en deux parties, une partie d'encodage et une partie de décodage. Quantification scalaire, qui relie une variable continue à une variable discrète qui peut prendre un nombre plus petit et fini de valeurs.

Les techniques de compression d'images exploitent souvent la redondance statistique présente dans les images.

- Principe de la quantification vectorielle

La quantification vectorielle dans son sens le plus général est l'approximation d'un signal d'amplitude continue par un signal d'amplitude discrète. Elle peut être vue comme une application Q associant à chaque vecteur d'entrée x de dimension K un vecteur $y = Q(x)$ de même dimension appartenant à un ensemble fini Y appelé DICTIONNAIRE de taille finie N , $Y = (y_j, j = 1...N)$.

Elle se décompose en deux applications : codeur, décodeur (cf. figure II.6) :

- Codeur

Le rôle du codeur consiste, pour tout vecteur x_i du signal en entrée à rechercher dans le dictionnaire Y le code vecteur y_j le plus proche du vecteur source x . C'est uniquement l'adresse du code vecteur y_j ainsi sélectionnée qui sera transmise ou stockée. C'est à ce niveau donc que s'effectue la compression.

- Décodeur

Il dispose d'une réplique du dictionnaire et consulte celui-ci pour fournir le code vecteur d'indice correspondant à l'adresse reçue. Le décodeur réalise l'opération de décompression.

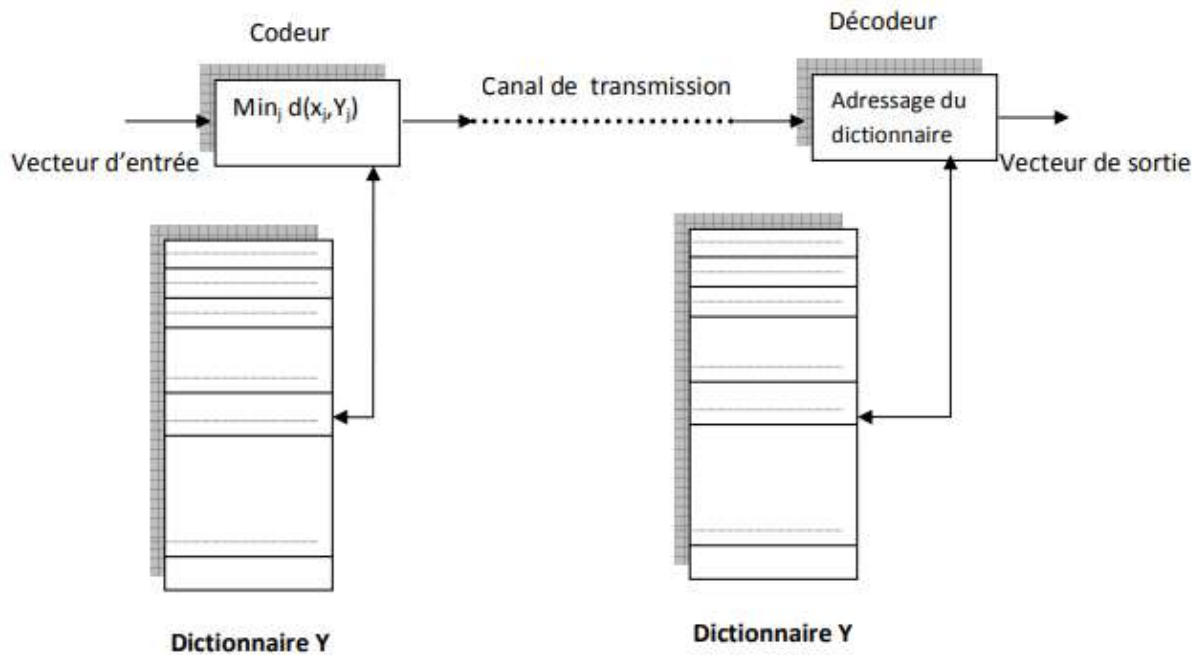


Figure II.6 Principe du Quantificateur vectoriel

Où $\text{Min}_j(x,y_j) =$ distorsion minimale entre les vecteurs x et y_j .

➤ 4)2.2 Méthodes par transformée

Dans ces méthodes, une image de taille $N \times N$ est subdivisée en sous-images ou blocs de taille réduite (le calcul nécessaire pour effectuer la transformation sur toute l'image est très élevé). Chaque bloc subit une transformation mathématique orthogonale linéaire inversible du domaine spatial vers le domaine fréquentiel, indépendamment des autres blocs (transformés en un ensemble de coefficients plus ou moins indépendants). Il s'agit plutôt de trier les coefficients pour faciliter le travail de l'encodeur entropique ou d'exploiter les dépendances entre les coefficients. Les coefficients obtenus sont ensuite quantifiés et codés pour rendre compte de leur transmission ou stockage. (9) (1) (5)

Dans une transformation linéaire existante :

- Transformée de Karhunen-Loeve (TKL).
- Transformée de Fourier discrète (DFT).
- Transformée d'Hadamard (TH).
- Transformée discrète en cosinus (DCT).
- Transformée en ondelettes (TO).

Un schéma typique de compression par transformée est représenté sur la Figure (II.7).

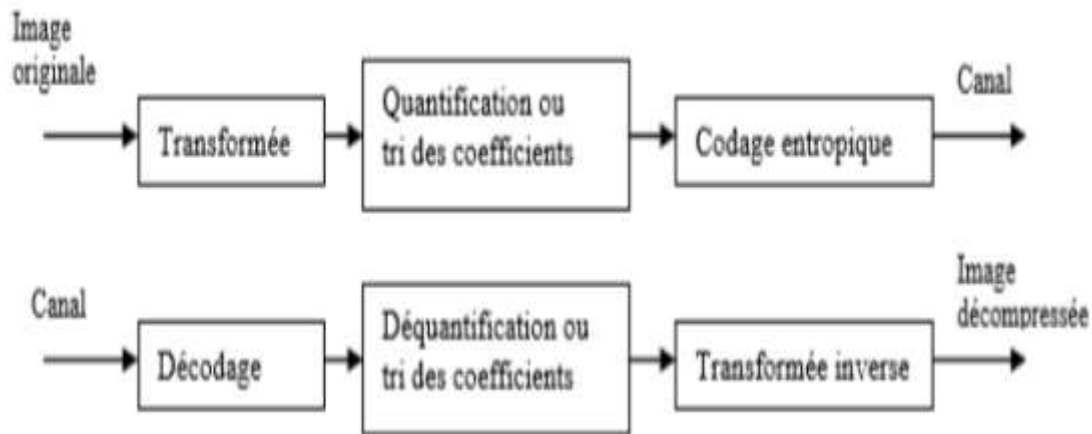


Figure II.7 Schéma de Compression –Décompression par transformée

Le but de la transformation est double, c'est :

- Décorrélérer les données, c'est-à-dire obtenir des coefficients de transformation moins corrélés que les pixels de l'image.
- Concentre l'énergie sur un plus petit nombre de coefficients qui ont des valeurs plus importantes aux basses fréquences qu'aux hautes fréquences.

❖ 4)2.2.1 Transformée en cosinus discrète (DCT) :

Son but est de regrouper les énergies en passant des notions de pixel et de couleur aux notions de fréquence et d'amplitude.

DCT convertir un signal d'amplitude discret bidimensionnel (chaque valeur du signal représente l'amplitude d'un phénomène) en information de "fréquence" bidimensionnelle, les équations suivantes donnent respectivement les transformées en cosinus discrètes directes et inverses. Quant aux hautes fréquences, elles sont situées dans le coin inférieur droit de la matrice, et plus on se rapproche du coin inférieur droit de la matrice, plus leurs coefficients tendent vers 0.

Après avoir appliqué cette transformée au bloc, nous pouvons stocker les informations fondamentales dans la base de fréquence et les énergies seront regroupées dans le coin supérieur gauche de la matrice.

Pour les images couleur, trois fonctions tridimensionnelles (indépendantes) sont traitées : X et Y, représentant le pixel et Z représentant la valeur du pixel à cet instant.

C'est l'acte d'appliquer une transformée en cosinus discrète à des blocs de pixels pour supprimer la redondance des données d'image. (9) (1) (5) (10) (8)

Cette transformation sera appliquée à chaque composant de chaque bloc. En pratique, pour une image codée en 24 bits, on appliquerait la DCT 3 fois par bloc (1 pour la composante Y, 1 pour Cb ou U, et 1 pour V ou Cr) (Figure II.8).

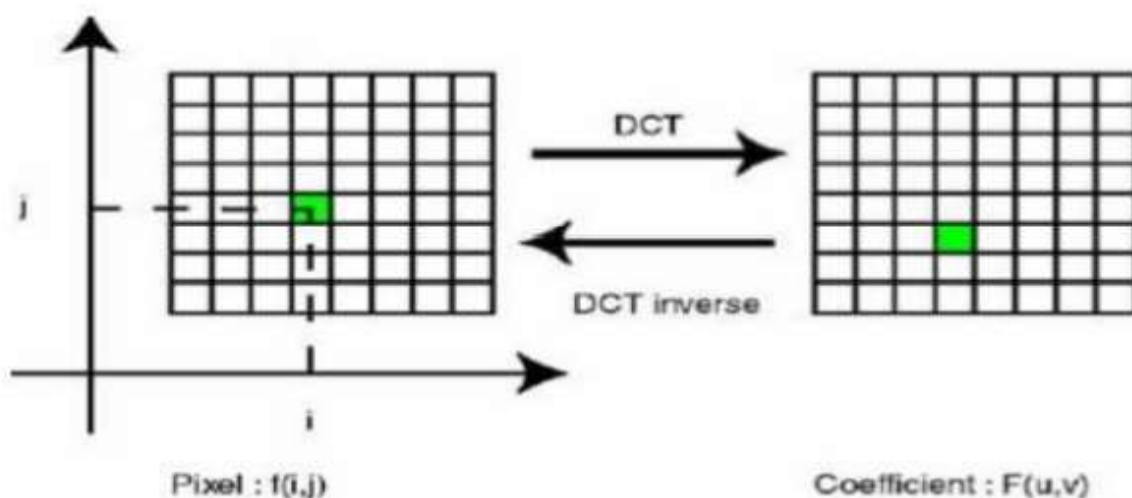


Figure II.8 Application de la DCT sur chaque bloc

La DCT s'exprime mathématiquement comme suite :

$$DCT(i, j) = \frac{1}{\sqrt{2n}} c(i) c(j) \sum_{x=0}^{n-1} \sum_{y=0}^{n-1} p(x, y) \cos\left(\frac{(2x+1)i\pi}{2n}\right) \cos\left(\frac{(2y+1)j\pi}{2n}\right)$$

- n : la largeur d'un bloc, ici n=8.
- i, j : les indices d'un coefficient de la DCT dans un bloc.
- x, y : les indices d'un pixel de l'image dans un bloc.
- DCT (i,j) : la valeur d'un coefficient dans un bloc.
- $c(x) = \frac{1}{\sqrt{2n}}$ si $x = 0$, $c(x) = \frac{1}{\sqrt{2}}$ si $x = 0$
1 ailleurs
- p(x,y) est la valeur du pixel aux coordonnées(x,y).

Les avantages de la DCT	Les inconvénients de DCT
Efficacité mathématique (facile à mettre en œuvre).	Elle besoin de diviser le codage d'entrée en bloc 2D non chevauchants.
Performance de compression élevée.	Mauvaise transformation de l'ensemble de l'image.
Sortie réelle.	Le rapport de compression est bas.
Transformation rapide (surtout quand il est utilisé dans une image compressée).	

Tableau II.3 Les avantages et Les inconvénients de DCT

❖ 4)2.2.1 Transformée en ondelettes discrètes (DWT) :

Reconstruction à l'aide d'un filtre spéculaire en quadrature (QMF),

Représentée par leurs réponses impulsionnelles (h et g), cette opération est l'inverse de l'opération de décomposition (remplacement des filtres passe-bas et passe-haut par les filtres miroirs associés), et l'opération de décimation est remplacée par l'opération d'interpolation incluse dans chacun. Ajouter un zéro entre les coefficients. Selon cela, le schéma de suralimentation est un algorithme efficace pour mesurer la DWT car il utilise moins de coefficients de filtre, ce qui permet une mise en œuvre rapide et une réduction de la mémoire de la DWT.

La transformée en ondelettes discrètes (DWT) fournit une représentation d'image multi-résolution et est devenue l'un des outils les plus importants dans l'analyse et le codage d'images au cours des deux dernières décennies.

La transformée en ondelettes discrètes (DWT) est une puissante transformée qui offre une efficacité de compression nettement meilleure que la DCT à des taux de compression plus élevés. Il présente des propriétés d'adaptabilité ou de flexibilité très intéressantes, notamment le choix des fonctions d'ondelettes de base (orthogonales ou sans support, compact ou infini, etc.) et des paramètres.

La transformée phare de la compression d'images de ces dernières années est la transformée en ondelettes discrètes (DWT Discrete Wavelet Transform).

La DWT inspire également de nombreuses autres transformées (paquets d'ondelettes, curvelets, stripsets, contourlets, etc.) pour la compression ou le débruitage d'image, cependant, les performances peuvent être améliorées en choisissant une base de transformée qui s'adapte à l'image. (1) (2) (5) (10)

➤ 4)2.3 Méthodes prédictives

L'idée de prédiction consiste à approximer un pixel ou un bloc d'une image à partir de données issues d'une portion de l'image courante qui a été encodée, résultant de la redondance spatiale de l'image, ou d'images préalablement décodées d'une séquence de travail. Redondance temporelle des données d'autres images. Le codeur calcule une prédiction pour le macrobloc courant sur la base de données codée provenant de la même image (prédiction intra) ou d'autres images (prédiction inter). Dans le codage prédictif, la valeur de chaque pixel est prédite à partir des pixels précédemment codés.

La méthode de prédiction est l'une des plus anciennes et c'est une méthode de décorrélation qui fonctionne comme suit. Les méthodes prédictives peuvent être facilement mises en œuvre et conduire à de bons taux de compression. (1)

➤ 4)2.4 Méthodes hybrides

Le terme hybride fait référence aux techniques qui combinent le codage prédictif et le codage par transformation.

Dans le cas d'images fixes, une transformation unidimensionnelle est effectuée le long des lignes, puis une prédiction est effectuée le long des colonnes. Pour les images animées, une combinaison est effectuée entre une transformation bidimensionnelle dans le domaine

spatial et une prédiction le long des composantes temporelles pour exploiter la redondance temporelle du signal image.

Un codeur hybride combine les avantages des deux techniques qui le composent.

II. 5) Les normes de compression des images avec et sans pertes

II. 5)1. La norme de compression JPEG

Après une longue période de recherche et développement, avec la participation d'une très large communauté scientifique, JPEG est devenu une norme internationale en 1993.

La norme JPEG (Joint Photographic Experts Group) a été conçue par le groupe ISO (International Standards Organization) et le groupe IEC (International Electronics Commission). Il est conçu pour compresser des images fixes en couleur et en niveaux de gris pour les stocker sur un support numérique. Il est conçu pour couvrir les applications les plus diverses, tout en prenant en compte les contraintes du monde réel associées aux applications les plus évidentes : édition, transmission, référentiels d'images. (2) (3) (5) (7) (8)

Les techniques définies par la norme JPEG se divisent en deux classes :

- les méthodes de compression avec pertes qui sont basées sur la TCD suivie d'une quantification et d'un codeur entropique (cf. figure II.9).
- La seconde classe, concerne les processus de codage sans pertes, cette classe de codeurs n'est pas basée sur la TCD mais sur le codage MICD suivi d'un codage entropique (cf. figure II.10). Pour les méthodes avec pertes, quatre codeurs ont été spécifiés : un codage de base où l'image compressée puis décompressée n'est plus identique à l'image originale, ce processus utilise la TCD et un codage de Huffman.

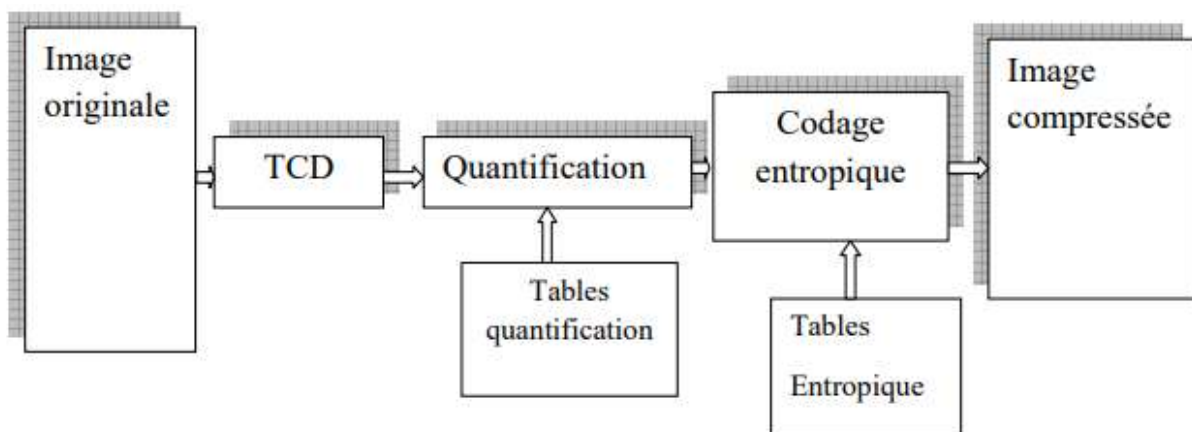


Figure II.9 Principe de l'algorithme JPEG avec pertes

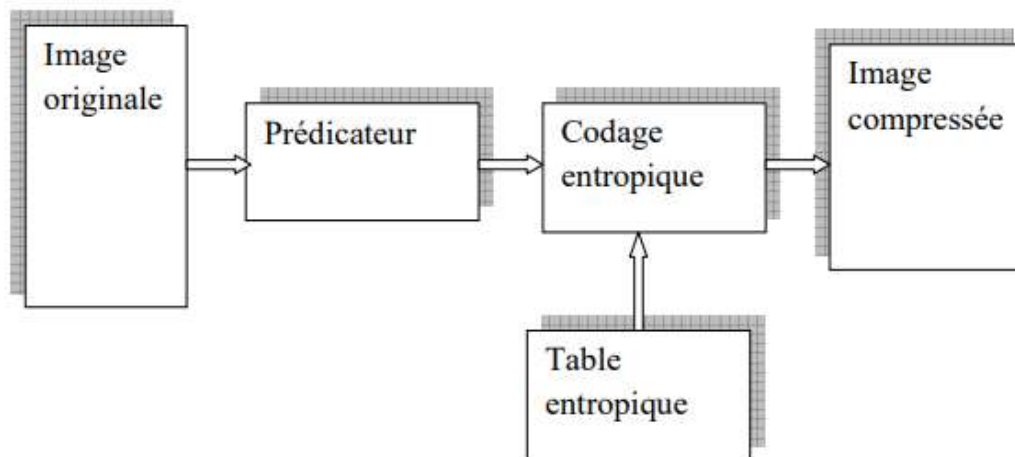


Figure II.10 Principe de l'algorithme JPEG sans pertes

Discrete Cosine Transform via 8x8 Blocks (DCT) : L'objectif du calcul DCT est de supprimer les hautes fréquences, et il y a 2 étapes de base pour atteindre cet objectif JPEG :

1. Séparer les basses et les hautes fréquences présentes dans une image : sauf Imagez en blocs et appliquez sa distribution spectrale à chaque pixel de bloc via DCT.

L'image est subdivisée en blocs de taille 8x8.

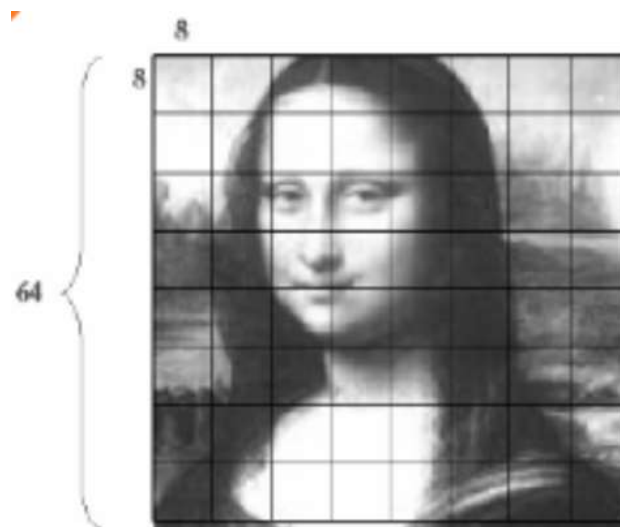


Figure II.11 Division de l'image en sous image 8*8

CHAPITRE II : Généralités sur les techniques de compression d'images

100	155	131	116	151	135	131	211
120	135	127	88	155	131	155	179
120	135	151	100	179	116	155	167
120	155	151	108	191	112	155	179
135	151	135	210	197	112	179	179
120	151	155	151	151	116	179	179
135	151	167	167	151	151	167	171
150	151	179	151	151	131	155	167

DCT

145	-84	34	-69	4	-66	35	72
-45	-28	28	19	10	-54	5	15
0	-2	-8	-15	-9	0	30	-41
9	-14	15	-11	5	8	-12	-32
1	1	3	-11	7	-23	-4	0
18	4	-17	-10	4	-10	7	-10
-5	1	-7	-20	1	-1	-3	5
3	1	1	9	2	7	2	-2

Les basses fréquences se trouvent en haut à gauche de la matrice, et les hautes fréquences en bas à droite.

2. Étape de compression : quantifier ces blocs transformés avec plus ou moins de bits selon la qualité requise

La quantification : La matrice de quantification est définie après que l'utilisateur a sélectionné la qualité de compression souhaitée.

Ce choix a été fait suite à des tests de facteurs de dégradation d'image associés à l'étape de quantification : les développeurs du format JPEG ont estimé qu'il ne fallait pas dépasser un facteur de 25, de sorte que

$$\text{Quantification}[i][j] = 1+(i+j+1) * \text{qualité}$$

Exemple : pour un facteur de qualité égale 3

Quantification=

4	7	10	13	16	19	22	25
7	10	13	16	19	22	25	28
10	13	16	19	22	25	28	31
13	16	19	22	25	28	31	34
16	19	22	25	28	31	34	37
19	22	25	28	31	34	37	40
22	25	28	31	34	37	40	43
25	28	31	34	37	40	43	46

Chaque coefficient du calcul DCT sera alors quantifié à l'aide de cette matrice.

$$\text{Coefficient [ligne] [colonne]} = \text{ROUND} \left(\frac{\text{Coefficient[ligne][colonne]}}{\text{Quantification[ligne][colonne]}} \right)$$

Compression RLE et Huffman : La dernière étape de l'algorithme JPEG consiste à utiliser des méthodes de compression classiques sans perte.

RLE : est par conséquent utilisé pour les séquences formées essentiellement de 0.

Huffman : pour compresser les autres valeurs des ensembles.

Le codage RLE est appliqué en premier d'une manière particulière pour les séquences de 0.

Décompression :

Nous avons vu que la compression JPEG se fait en 5 étapes. Pour décompresser l'image, il faut donc prendre le chemin inverse. On part d'un fichier contenant une séquence de bits : ceux-ci doivent contenir des facteurs de compression (reconstruction des tables de quantification), des tables de codage Huffman (conversion du codage Huffman de DCT en coefficients).

Une fois avoir lu ces informations dans l'entête, on procède à la décompression.

- 1) Reconstruction des coefficients à partir de la table de Huffman.
- 2) Repérer la balise du codage RLE pour remettre les bonnes séquences de zéros.
- 3) Procéder à une transformation par bloc 8x8 de type IDCT (Inverse Discrète Cosinus Transform).

La formule pour Calculer la IDCT sur une matrice NxN :

$$\text{Pixel}(x, y) = \frac{1}{\sqrt{2n}} \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} c(i)c(j)DCT(i, j) \cos\left(\frac{(2x+1)i\pi}{2n}\right) \cos\left(\frac{(2x+1)j\pi}{2n}\right)$$

II. 5)2. La norme de compression JBEG2000

L'un des principaux inconvénients de JPEG est la perte de netteté des bords en utilisant une quantification qui a tendance à lisser l'image, donc JPEG est nul à des taux de compression très élevés. La norme JPEG a parfois eu du mal à répondre aux besoins identifiés par ces applications, ce qui a conduit à la création d'une nouvelle version de la norme ; JPEG 2000, pour intégrer des technologies plus performantes et répondre à ces nouveaux besoins.

Des applications relativement inexploitées lors de la définition de la norme JPEG occupent désormais une place prépondérante dans l'imagerie numérique : Internet, la médecine, les appareils photo numériques, la transmission d'images sur les réseaux mobiles, etc.

La technologie d'aujourd'hui n'est plus la même, car la recherche a évolué et, d'autre part, les besoins ne sont plus les mêmes.

JPEG 2000 est une évolution du format JPEG qui va reprendre certains éléments du JPEG, comme le prétraitement des images en passant du RVB, nécessaire en réponse aux évolutions de la photographie numérique, du web et de l'e-commerce.

Deux types de méthodes de compression par ondelettes sont aujourd'hui référencés : les algorithmes de codage imbriqués {EZW (Embedded Zerotree Wavelet), SPIHT (Set Partitionning In Hierarchical Trees).

CHAPITRE II : Généralités sur les techniques de compression d'images

La norme JPEG2000 est basée sur la transformée en ondelettes discrètes DWT au lieu de la DCT utilisée par la norme JPEG.

Le but de ces transformations est généralement de décorrélérer les données brutes de l'image représentée par ses pixels. DWT permet une analyse plus localisée et plus fine des informations, ce que DCT ne peut pas.

Un tel programme est difficile à réaliser. Le groupe de travail ISO JPEG 2000 y travaille depuis 1998. (2) (5) (7) (8)

On peut tout de même donner un premier aperçu de ce que devrait être le format JPEG 2000 en présentant 3 de ses caractéristiques principales:

- Il sera basé sur la technologie des ondelettes. Le résultat pratique de ce choix devrait être une amélioration modeste, mais réelle, du taux de compression (environ 20%) et de la qualité de reproduction par rapport à ce que permet le JPEG actuel. Les coefficients d'ondelettes sont plus faciles à stocker que les blocs produits en appliquant DCT.
- Il permettra à l'utilisateur de déterminer jusqu'à quel point les détails de l'image l'intéressent en lui donnant le choix de la résolution de restitution. La récupération sans pertes de l'image devrait être possible.
- Il permettra une meilleure restitution des couleurs grâce à l'utilisation des profils ICC. On pourra aussi bien enregistrer des images RVB que des images CMJN.

II. 5)3. La norme de compression JBIG

JBIG est une norme de compression sans perte adaptée aux images à deux tons seulement (images à deux couches, telles que les images de télécopie).

La norme s'applique également aux débits binaires de 9,6 Kbits/S à 64 Kbits/S. Il utilise un codage sans perte avec un codeur arithmétique adaptatif.

La norme JBIG (Joint Bilayer Image Group) est conçue pour compresser des images photographiques représentées sous forme d'images textuelles bicolores (noir et blanc). Sa structure est en couches et chaque couche est un encodeur indépendant. (3)

II. 6) Codages imbriqués

II. 6)1. Algorithme EZW de Shapiro

Par rapport à la norme JPEG, l'encodeur EZW offre les caractéristiques de transfert progressif de l'image encodée tout en offrant des performances supérieures (débit-distorsion). La structure zerotree peut détecter les régions d'une image qui ne contiennent pas d'informations importantes, puis les encoder dans un arbre.

L'EZW (Embedded Zerotree Wavelet) proposé par J. Shapiro a été le premier encodeur de sous-bande zerotree à être introduit.

L'algorithme EZW peut être résumé en trois étapes, comme suit :

CHAPITRE II : Généralités sur les techniques de compression d'images

- La définition des cartes de signifiante indiquant les positions des coefficients significatifs par rapport à un seuil donné.
- Une approximation successive, par passes, des coefficients significatifs, qui permet donc une notion de progressivité du codage selon un critère d'arrêt de débitdistorsion.
- Un codeur arithmétique dynamique de la chaîne de symboles.

L'algorithme EZW a été ensuite amélioré par l'algorithme SPIHT et SPECK. (1) (2) (3)

II. 6)2. Algorithme SPIHT

Sur les coefficients des sous-bandes, l'algorithme SPIHT applique les règles de partitionnement définies. Les algorithmes du codeur et du décodeur sont les mêmes et aucune information d'ordre de transmission explicite n'est requise. Il est le même que d'autres algorithmes de transmission progressive pour l'in- codage du véhicule, ce qui améliore l'efficacité de codage de l'algorithme par rapport à d'autres algorithmes. Ainsi, le flux de sortie contient les informations générées par l'algorithme de tri, les informations nécessaires au décodeur pour pouvoir réorganiser (d'érier) les coefficients, et les informations liées aux valeurs des coefficients (signe et bits les moins significatifs).

Ainsi, le tri partiel utilise une recherche en profondeur des nœuds qui ont des descendants significatifs dans l'arbre hiérarchique, réorganisant les coefficients des branches en fonction de leurs bits les plus significatifs (réduisant l'erreur quadratique). Par conséquent, des coefficients insignifiants dans les basses fréquences sont probablement dus à des enfants insignifiants dans les hautes fréquences, et ceux qui ont une énergie élevée ont généralement des enfants avec moins d'énergie. Cependant, les coefficients sont partiellement ordonnés à l'aide d'une arborescence hiérarchique de coefficients pour coder préférentiellement les coefficients à haute énergie. Il est basé sur le même concept : codage progressif à travers des plans de bits et utilisant des dépendances hiérarchiques entre différents coefficients de sous-bande.

En 1996, Said et Pearlman ont créé un algorithme de codage d'images simple et entièrement intégré basé sur le principe de la division des ensembles en arbres hiérarchiques (SPIHT).

L'algorithme SPIHT permet la génération directe de flux binaires au lieu des symboles significatifs utilisés dans EZW.

Comme EZW, cet algorithme suppose une décroissance énergétique pour les coefficients de résolution/fréquence la plus élevée dans l'arbre hiérarchique.

Cependant, un nouveau protocole de dépendance entre les coefficients est défini. Les coefficients de la sous-bande de plus basse fréquence (A2 en figure II.12) sont regroupés par quatre, et pour chaque groupe, la descendance est comme suit :

Un des quatre coefficients (marqué par « * » en figure II.12) n'admet pas des descendants. Pour les autres sous-bandes, comme dans le cas de l'EZW, chaque coefficient admet quatre descendants. (1) (2) (3) (10)

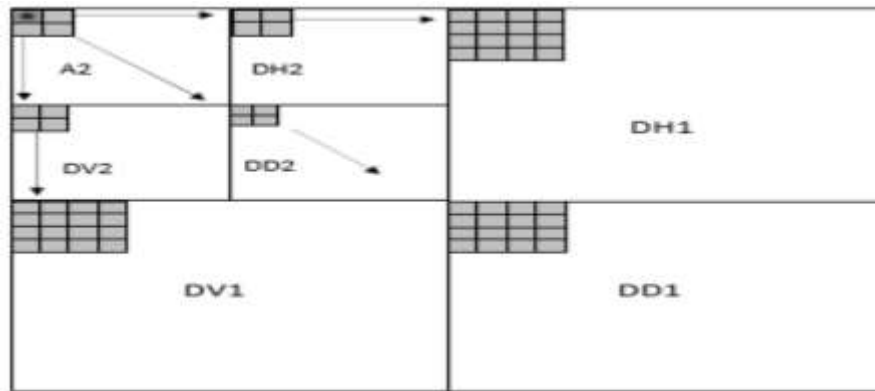


Figure II.12 Exemples de descendance parent-fils dans le cas SPIHT, le pixel désigné par (*) n'accepte pas de descendants

Trois ensembles de coordonnées sont utilisés pour présenter la nouvelle méthode de codage :

$D(i,j)$: Ensemble de tous les descendants du pixel (i,j) .

$O(i,j)$: Ensemble de coordonnées des descendants directs du pixel (i,j) .

$L(i,j)$: Ensemble défini comme étant $D(i,j) - O(i,j)$.

II. 6)3. Algorithme SPECK

L'algorithme SPECK offre des performances comparables à l'algorithme SPIHT en exploitant la structure d'ensembles de coefficients sans signification dans des blocs plutôt que des arbres. Ces structures de blocs permettent de s'affranchir efficacement de la non-stationnarité des coefficients (ordre 1) en ajustant localement les statistiques utilisées pour le codage. (2) (3)

II. 6)4. Algorithme EZBC

Le principe de codage de l'algorithme EZBC (Embedded ZeroBlocks coding based on Context modeling) est similaire à l'algorithme SPECK. L'innovation de cet encodeur vient principalement de l'exploitation des dépendances entre les nœuds de l'arbre quaternaire d'importance. Le partitionnement Quadtree est également effectué indépendamment dans chaque sous-bande, permettant une meilleure séparation des statistiques de saillance et un apprentissage plus efficace avec un contexte plus large.

Les performances de cet encodeur sont comparables à EBCOT, et il convient également à l'encodage vidéo appelé MC-EZBC. (2) (3)

II. 7) Conclusion

Notez que chaque encodeur a des avantages et des inconvénients, donc aucun encodeur général n'est utilisé dans l'opération de compression, nous pouvons donc en déduire que changer l'image doit changer l'algorithme de compression.

CHAPITRE II : Généralités sur les techniques de compression d'images

Afin d'obtenir une meilleure qualité d'image, certaines méthodes et mesures sont introduites pour supporter le procédé de compression selon le domaine d'utilisation du procédé de compression. Son importance tient avant tout aux possibilités matérielles des appareils que nous utilisons (capacité de mémoire de masse, débit internet, etc.) et aux besoins exprimés par les utilisateurs (visiophonie, vidéo plein écran, transmissions de masse obtenues dans des délais de plus en plus courts périodes de temps) Plus d'informations. L'avenir de la compression ne peut être atteint que grâce à des algorithmes mathématiques, et les progrès de la recherche mathématique conduiront inévitablement à des progrès dans la compression d'images, et il existe de multiples formats qui compensent la diversité des images, leur taille, leur nombre des couleurs, et représentation (vecteur et bitmap).

Au final, nous pouvons conclure qu'il existe de nombreuses façons d'encoder, de représenter et de compresser des images. Les ratios deviennent de plus en plus grands sans perte notable et réelle de qualité d'image.

En raison du développement du réseau et du multimédia, la compression des images jouera un rôle plus important.

Aujourd'hui, avec les énormes avancées technologiques, les algorithmes se permettent d'effectuer de nombreux calculs pour compresser et décompresser les images.

Le chapitre 3 fournira un aperçu des sujets de taxonomie et de ces domaines d'application.

CHAPITRE III
Résultats expérimentaux

III. 1) Introduction :

Dans ce chapitre, nous étudions les résultats obtenus par notre méthode sur les images tests utilisées, et nous définirons le protocole de test pour valider notre méthode.

Les métriques utilisées sont : PSNR et SSIM.

III. 2) L'environnement de développement :

MATLAB : est une plate-forme de programmation conçue spécifiquement pour les ingénieurs et les scientifiques afin d'analyser et de concevoir des systèmes et des produits qui transforment notre monde. Le cœur de MATLAB est le langage MATLAB, un langage matriciel permettant l'expression la plus naturelle des mathématiques computationnelles. (15)



Figure III.1 : Icône MATLAB

III. 3) Technique de compression :

Nous avons deux techniques de compression que nous avons mentionnées précédemment, à savoir la compression d'image sans perte de données et la compression d'image avec perte de données.

La compression d'image est une application de compression de données sur des images numériques.

L'organigramme :

La compression d'une image suit un certain nombre d'étapes visant à réduire la place occupée à l'aide de diverses méthodes. Ces étapes sont illustrées dans le schéma ci-dessous :

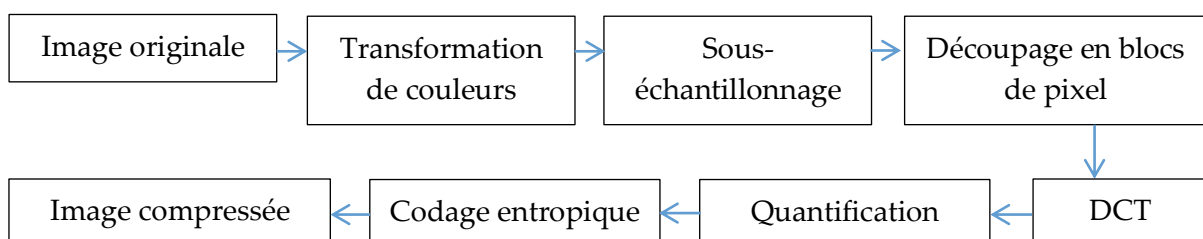


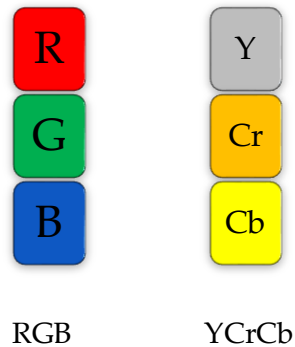
Figure III.2 : L'organigramme de la compression d'une image

CHAPITRE III : Résultats expérimentaux

Nous allons maintenant décrire chacune de ces étapes permettant la compression.

Transformation de couleurs:

La première étape consiste à passer du modèle initial des couleurs de l'image (souvent RVB) en modèle de type chrominance/luminance (YCrCb).



Ce modèle définit un pixel en fonction de sa luminance (intensité de la couleur) et de deux informations de chrominance (la couleur du pixel).

C'est l'étape la plus gourmande en temps et en ressources dans la compression et la décompression, mais c'est probablement la plus importante puisqu'elle nous permet de séparer les basses et les hautes fréquences présentes dans l'image.

Les meilleurs taux de compression sont obtenus avec un codage couleur de type luminance/chrominance (par exemple YUV, YCbCr), car l'œil humain est très sensible à la luminance, mais moins sensible à la chrominance.

Cette étape va permettre de préparer la prochaine étape : le sous-échantillonnage.

Le sous-échantillonnage :

L'objectif ici va être de réduire l'information occupée par la chrominance. Si l'œil humain est très sensible à la luminance, elle distingue par contre très mal la chrominance. Ainsi, une perte d'information sur ce point passera relativement inaperçu.

Le découpage en bloc de pixels :

À partir d'ici, la compression va travailler sur des blocs de pixels seulement. Ces blocs sont des matrices de pixels 8*8 soient 64 pixels.

$$f = \begin{bmatrix} 139 & 144 & 149 & 153 & 155 & 155 & 155 & 155 \\ 144 & 151 & 153 & 156 & 159 & 156 & 156 & 156 \\ 150 & 155 & 160 & 163 & 158 & 156 & 156 & 156 \\ 159 & 161 & 162 & 160 & 160 & 159 & 159 & 159 \\ 159 & 160 & 161 & 162 & 162 & 155 & 155 & 155 \\ 161 & 161 & 161 & 161 & 160 & 157 & 157 & 157 \\ 162 & 162 & 161 & 163 & 162 & 157 & 157 & 157 \\ 162 & 162 & 161 & 161 & 163 & 158 & 158 & 158 \end{bmatrix}$$

CHAPITRE III : Résultats expérimentaux

DCT :

La DCT (Discrete Cosine Transform) est une étape qui permet d'évaluer l'amplitude des changements d'un pixel à l'autre afin d'identifier les hautes et basses fréquences. Les basses fréquences, très présentes dans une image, sont des zones unies où les couleurs sont proches les unes des autres. A l'inverse, les hautes fréquences sont des zones de contraste, de changement rapide dans les couleurs.

Cette étape consiste à appliquer à notre matrice de pixels la formule mathématique du DCT afin d'obtenir une matrice des fréquences. Cette matrice sera utilisée dans la prochaine étape : la quantification.

$$f = \begin{bmatrix} 139 & 144 & 149 & 153 & 155 & 155 & 155 & 155 \\ 144 & 151 & 153 & 156 & 159 & 156 & 156 & 156 \\ 150 & 155 & 160 & 163 & 158 & 156 & 156 & 156 \\ 159 & 161 & 162 & 160 & 160 & 159 & 159 & 159 \\ 159 & 160 & 161 & 162 & 162 & 155 & 155 & 155 \\ 161 & 161 & 161 & 161 & 160 & 157 & 157 & 157 \\ 162 & 162 & 161 & 163 & 162 & 157 & 157 & 157 \\ 162 & 162 & 161 & 161 & 163 & 158 & 158 & 158 \end{bmatrix} \rightarrow F = \begin{bmatrix} 1260 & -1 & -12 & -5 & 2 & -2 & -3 & 1 \\ -23 & -17 & -6 & -3 & -3 & 0 & 0 & -1 \\ -11 & -9 & -2 & 2 & 0 & -1 & -1 & 0 \\ -7 & -2 & 0 & 1 & 1 & 0 & 0 & 0 \\ -1 & -1 & 1 & 2 & 0 & -1 & 1 & 1 \\ 2 & 0 & 2 & 0 & -1 & 1 & 1 & -1 \\ -1 & 0 & 0 & -1 & 0 & 2 & 1 & -1 \\ -3 & 2 & -4 & -2 & 2 & 1 & -1 & 0 \end{bmatrix}$$

La Quantification :

La quantification a pour objectif d'atténuer les hautes fréquences d'une image qui ont été mis en évidence par la DCT. L'œil humain distingue mal les zones de contrastes (les hautes fréquences) et la quantification va permettre de diminuer l'importance de ces informations superflues.

La technique est simple. Il suffit de diviser notre matrice des fréquences avec la matrice de quantification pour obtenir notre matrice quantifiée.

Cette matrice a un avantage qui sera utilisée dans la prochaine étape : les valeurs non nulles ne sont toutes situées dans le bord haut-gauche de la matrice.

$$F = \begin{bmatrix} 1260 & -1 & -12 & -5 & 2 & -2 & -3 & 1 \\ -23 & -17 & -6 & -3 & -3 & 0 & 0 & -1 \\ -11 & -9 & -2 & 2 & 0 & -1 & -1 & 0 \\ -7 & -2 & 0 & 1 & 1 & 0 & 0 & 0 \\ -1 & -1 & 1 & 2 & 0 & -1 & 1 & 1 \\ 2 & 0 & 2 & 0 & -1 & 1 & 1 & -1 \\ -1 & 0 & 0 & -1 & 0 & 2 & 1 & -1 \\ -3 & 2 & -4 & -2 & 2 & 1 & -1 & 0 \end{bmatrix} \rightarrow F^* = \begin{bmatrix} 79 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ -2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Le codage :

Souvent, le codage peut être transmis d'une représentation de données à une autre.

La dernière étape de la compression est l'encodage. Le but est d'encoder notre bloc de pixels.

III. 4) Tests et Résultats de la compression :

III. 4)1. Tests sur l'image standard :

Pour évaluer les performances de l'algorithme, nous avons effectué testé sur des images. Pour mieux comprendre la corrélation des résultats obtenus, deux grandeurs caractéristiques sont considérées. Il s'agit de l'indice de similarité structurelle (SSIM) et du rapport signal sur

CHAPITRE III : Résultats expérimentaux

bruit (PSNR). Ces derniers sont représentés par des pics et sont déterminés en comparant l'image reconstruite compressée/décompressée avec l'original.

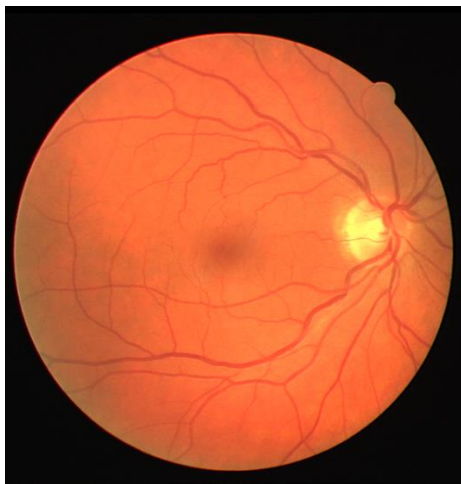
III. 4)2. Ensemble de données et ses caractéristiques:

1. 2_test.tif, dimension 565 x 584, taille 672 Ko.
2. 07_test.tif, dimension 565 x 584, taille 772 Ko.
3. 15_test.tif, dimension 565 x 584, taille 762 Ko.
4. 20_test.tif, dimension 565 x 584, taille 694 Ko.

III. 4)3. Formats d'image utilisés :

Pour notre application, nous avons utilisé des images testes en couleur (les images rétinienne)

a) 2_test.tif



b) 07_test.tif



c) 15_test.tif

CHAPITRE III : Résultats expérimentaux



d) 20_test.tif



Figure III.3 : Images testes utilisées

Ici, nous montrons quelques images de l'application :

Ceci est l'interface générale de l'application

CHAPITRE III : Résultats expérimentaux

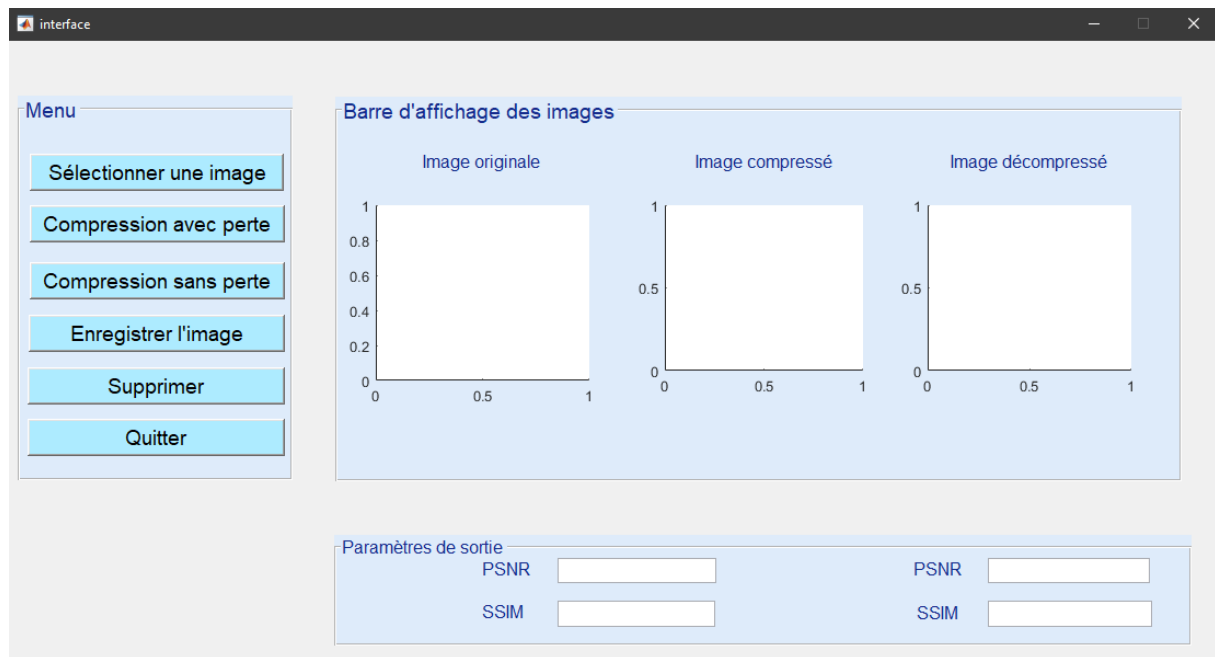


Figure III.4 : l'interface générale de l'application

Nous cliquons sur le bouton (sélectionner une image) pour choisir l'image à compresser.

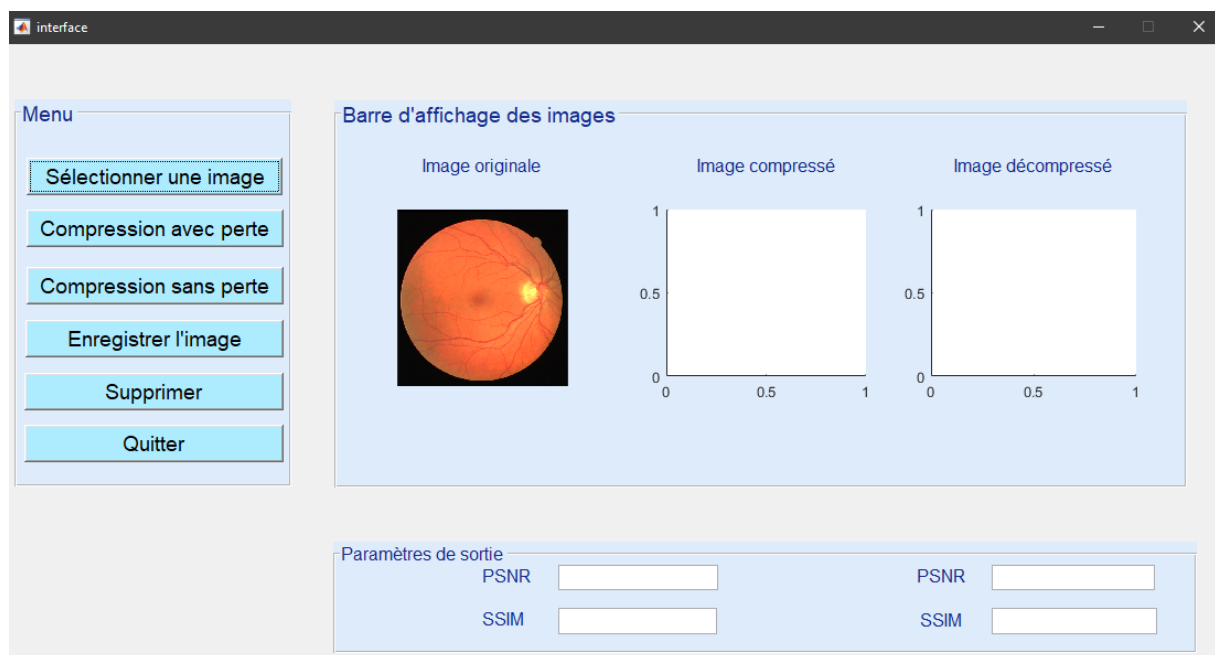


Figure III.5 : L'interface de l'image à compresser avec perte

Ensuite, nous appuyons sur le bouton (compression avec perte) pour voir le résultat.

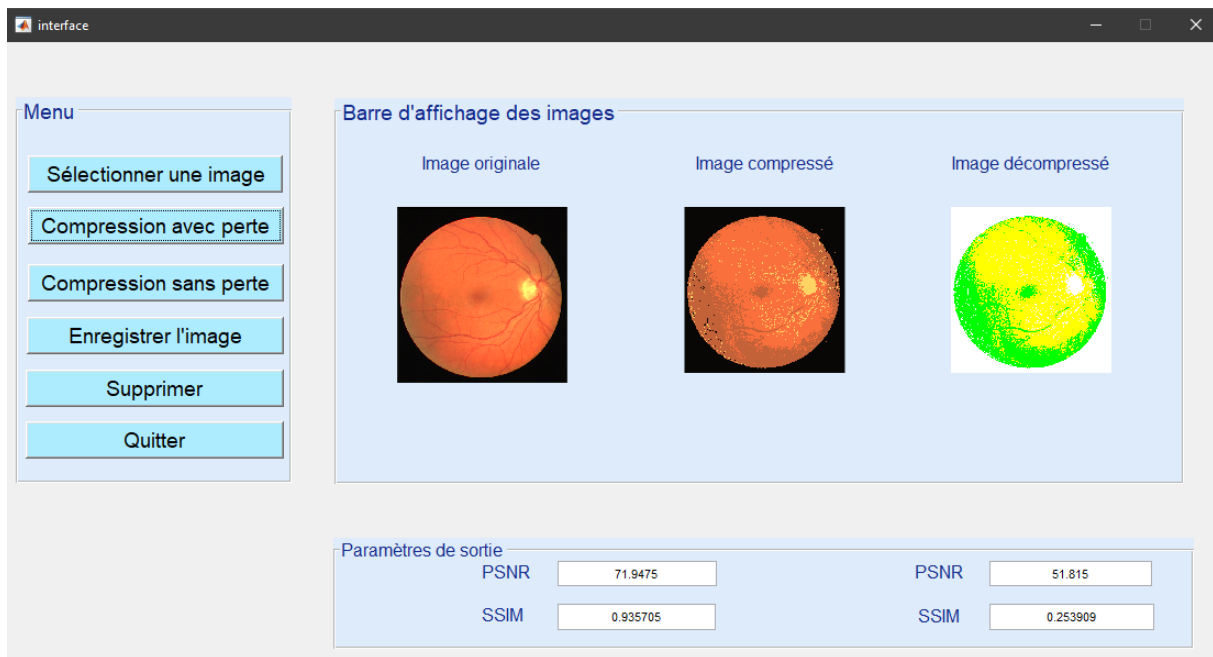


Figure III.6 : Interface de résultats de compression avec perte

Et ici la même chose pour la compression sans perte, où l'on passe par les mêmes étapes.

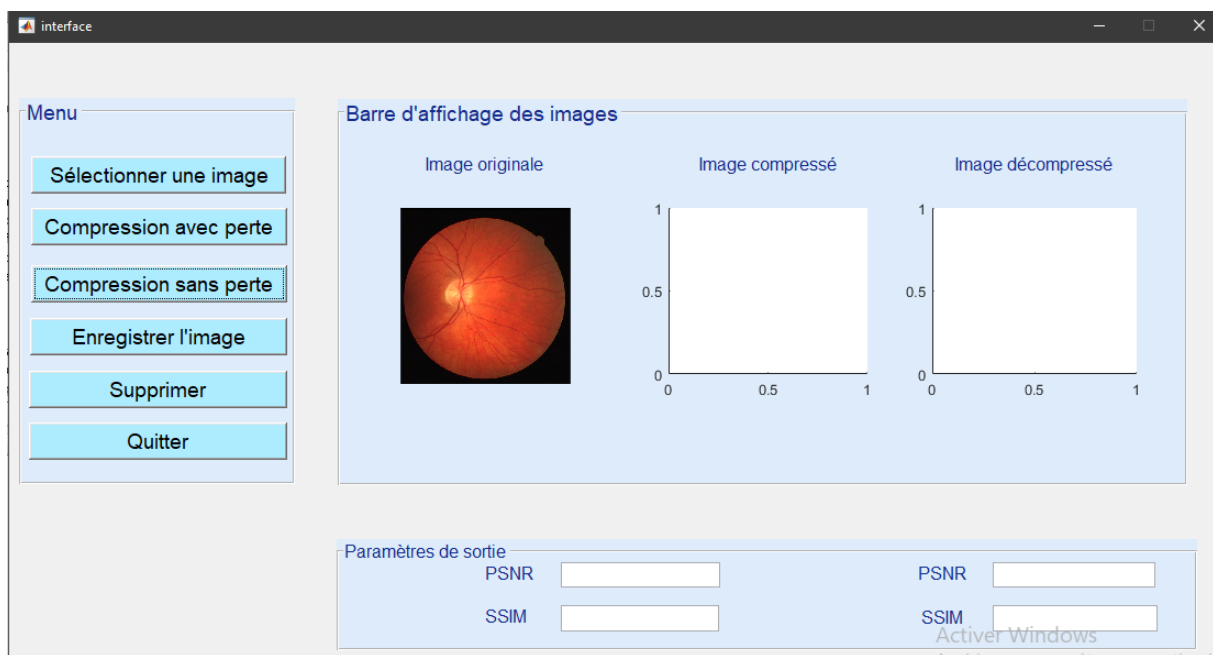


Figure III.7 : L'interface de l'image à compresser sans perte

CHAPITRE III : Résultats expérimentaux

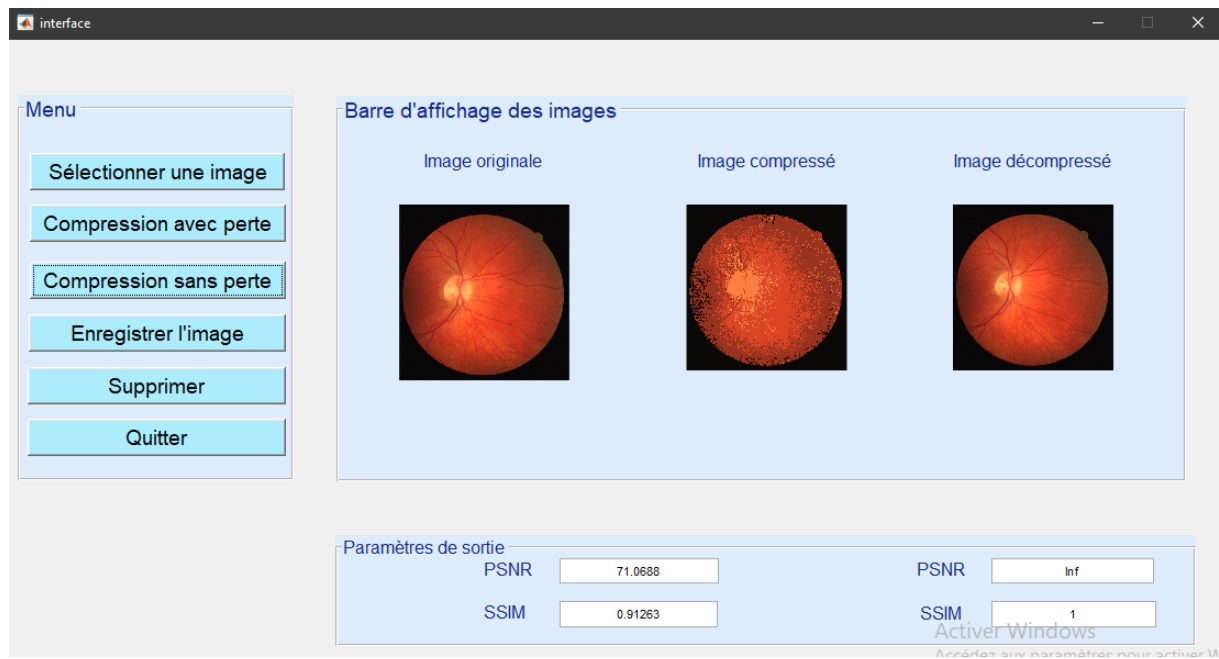


Figure III.8 : Interface de résultats de compression sans perte

Et ici si on veut supprimer l'image d'origine afin d'insérer une autre image.

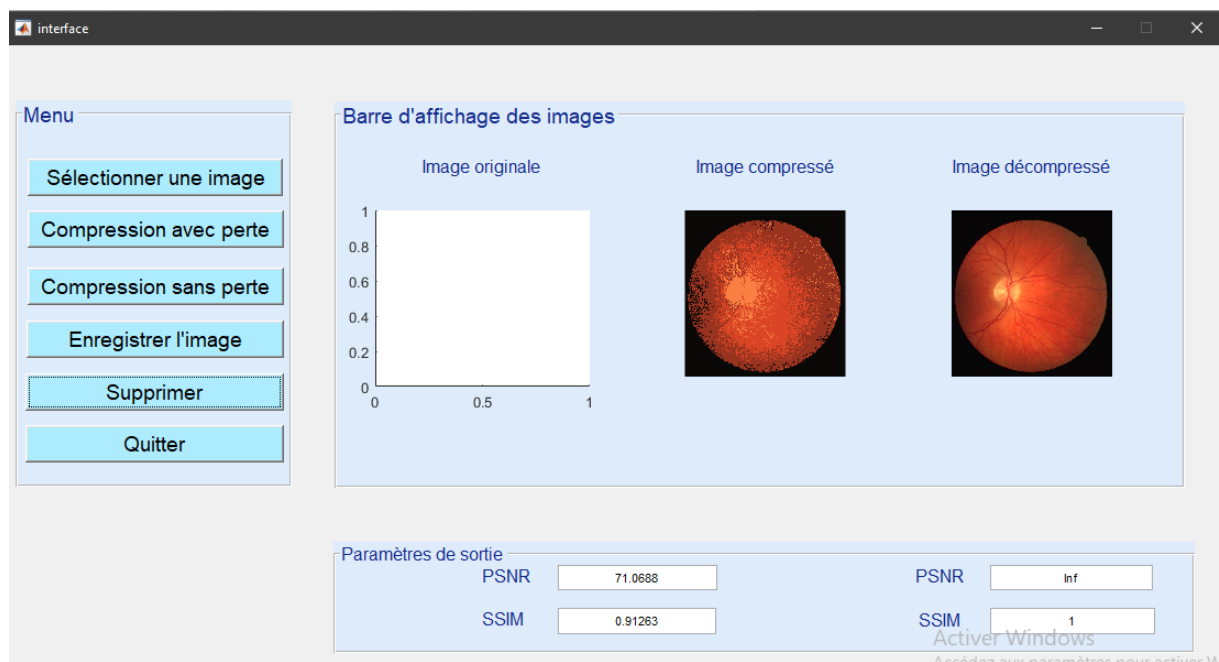


Figure III.9 : L'interface pour supprimer l'image d'origine

Et à la fin, si nous voulons quitter l'application, nous appuyons sur le bouton de sortie.

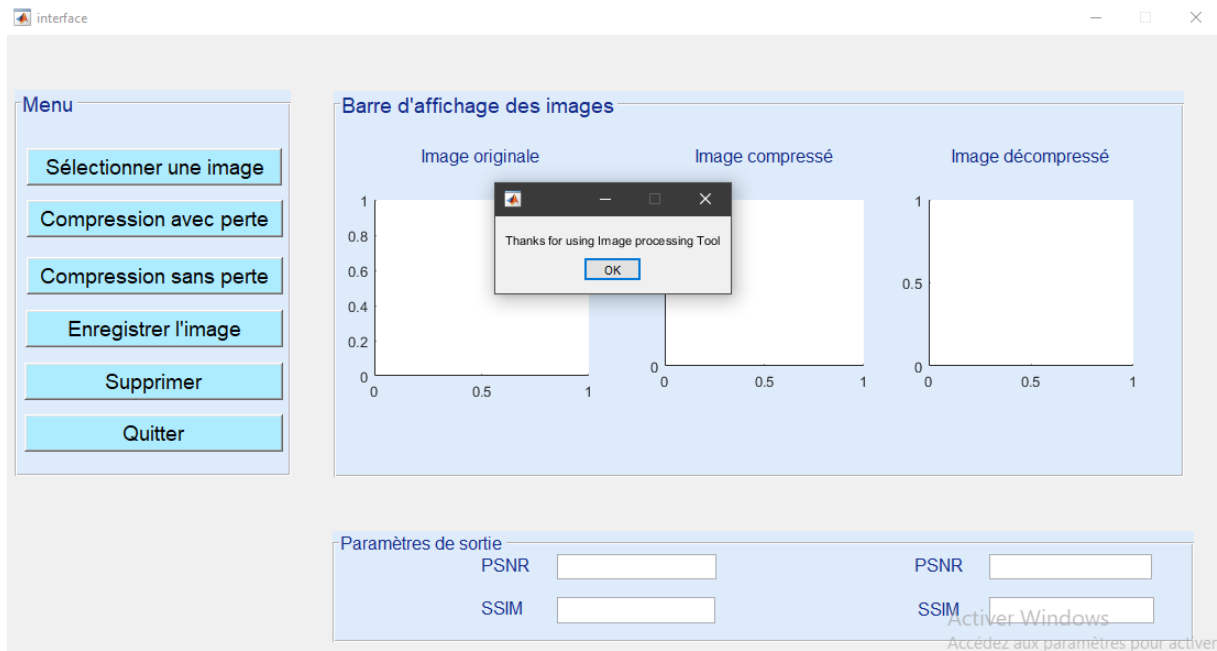


Figure III.10 : Interface de sortie

III. 4)4. Les résultats et interprétations :

Le rapport signal/bruit maximal (PSNR) et l'indice de similarité structurale (SSIM) sont deux outils de mesure largement utilisés dans l'évaluation de la qualité d'image. Surtout dans les images stéganographiques, ces deux instruments de mesure sont utilisés pour mesurer la qualité de la stéganographie. (1)

Mesure de la distorsion :

La distorsion entre l'image originale et l'image reconstituée est donnée par l'erreur quadratique moyenne (EQM) et le rapport signal sur bruit crête (PSNR) : (1) (2)

$$PSNR = 10 \cdot \log_{10} \left(\frac{(255)^2}{EQM} \right)$$

$$EQM = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2$$

- n : nombre de points de données
- Y_i : valeurs observées
- \hat{Y}_i : valeurs prédites

Structural Similarity :

Structural Similarity ou SSIM est une mesure de similarité entre deux images numériques.

Il a été développé pour mesurer la qualité visuelle des images compressées par rapport à l'original. L'idée de SSIM est de mesurer la similarité structurale entre deux images, et non la

CHAPITRE III : Résultats expérimentaux

différence pixel à pixel comme le PSNR. L'hypothèse de base est que l'œil humain est plus sensible aux modifications de la structure de l'image.

III. 4)4.1 Compression d'image sans perte :

Les images suivantes montrent les résultats de la compression sans perte :

2_test

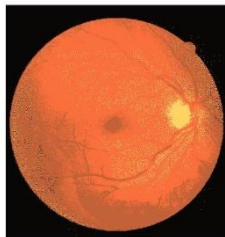


Figure III.11 : Image originale X

Figure III.12 : Image compressé Y

Figure III.13 : Image décompressé Z

07_test

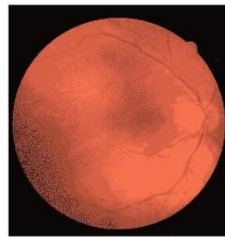


Figure III.14 : Image originale A

Figure III.15 : Image compressé P

Figure III.16 : Image décompressé C

15_test

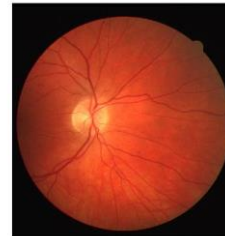
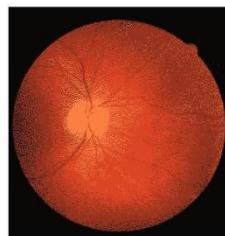


Figure III.17 : Image originale D

Figure III.18 : Image compressé E

Figure III.19 : Image décompressé F

CHAPITRE III : Résultats expérimentaux

20_test

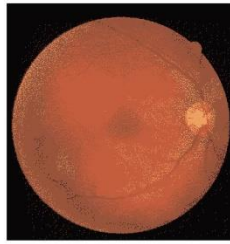
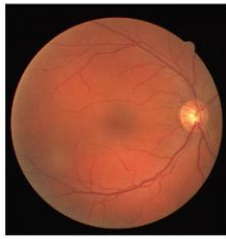


Figure III.20 : Image originale H **Figure III.21:** Image compressé I **Figure III.22:** Image décompressé K

Le tableau III.1 résume certains résultats des tests d'image. Nous avons utilisé le type de compression de données sans perte.

		2_test	07_test	15_test	20_test
PSNR et SSIM entre l'image originale et l'image compressé	PSNR	71.9475	72.2008	71.0688	73.2668
	SSIM	0.935705	0.918921	0.91263	0.87692
PSNR et SSIM entre l'image originale et l'image décompressé	PSNR	Inf	Inf	Inf	Inf
	SSIM	1	1	1	1

Tableau III.1 : Résultats des PSNR et SSIM (Compression sans perte)

IV. 4)4.2 Compression d'image avec perte :

Les images suivantes montrent les résultats de la compression avec perte :

2_test

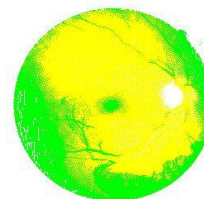
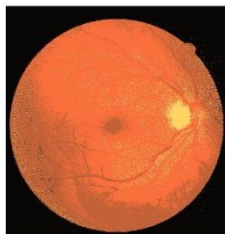


Figure III.23 : Image originale S **Figure III.24 :** Image compressé F **Figure III.25 :** Image décompressé B

CHAPITRE III : Résultats expérimentaux

07_test

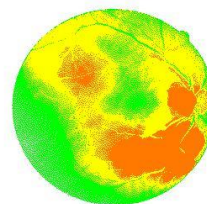
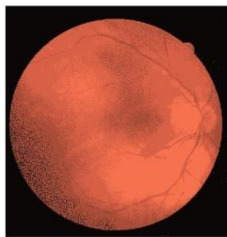


Figure III.26 : Image originale N

Figure III.27 : Image compressé M

Figure III.28 : Image décompressé W

15_test

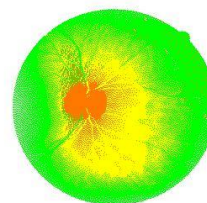
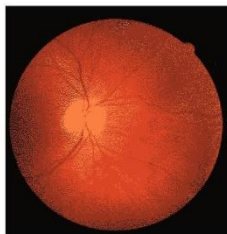


Figure III.29 : Image originale Q

Figure III.30 : Image compressé T

Figure III.31 : Image décompressé U

20_test

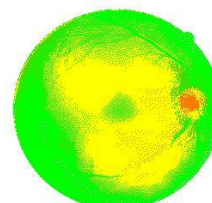
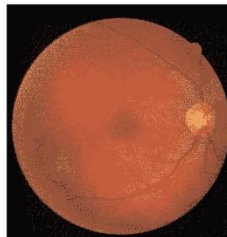


Figure III.32 : Image originale L

Figure III.33 : Image compressé J

Figure III.34 : Image décompressé V

Le tableau III.2 résume certains résultats des tests d'image. Nous avons utilisé le type de compression de données avec perte.

		2_test	07_test	15_test	20_test
PSNR et	PSNR	71.9475	72.2008	71.0688	73.2668
SSIM entre					
l'image					
originale et	SSIM	0.935705	0.918921	0.91263	0.87692

l'image compressé					
PSNR et	PSNR	51.815	51.6746	51.317	51.6948
SSIM entre					
l'image originale et	SSIM	0.253909	0.164136	0.087051	0.147141
l'image décompressé					

Tableau III.2 : Résultats des PSNR et SSIM (Compression avec perte)

Les résultats obtenus sont considérés satisfaisants, si nous prenons, par exemple, la compression sans perte des données et la compression avec perte des données.

III. 5) Évaluation et Objectif :

Le tableau III.3 résume certains résultats des tests d'image. Nous avons utilisé deux types de compression (compression de données sans perte/compression de données avec perte).

	PSNR	SSIM
2_test (Compression sans perte)	Inf	1
2_test (Compression avec perte)	51.815	0.253909
07_test (Compression sans perte)	Inf	1
07_test (Compression avec perte)	51.6746	0.164136
15_test (Compression sans perte)	Inf	1
15_test (Compression avec perte)	51.317	0.087051
20_test (Compression sans perte)	Inf	1
20_test (Compression avec perte)	51.6948	0.147141

Tableau III.3 : Résultats des PSNR et SSIM

Ces différents résultats montrent que le meilleur PSNR, celui dont la valeur est supérieure en compression, car le PSNR est supérieur en compression sans perte de données par rapport à la compression avec perte de données et la même chose pour le SSIM.

L'objectif de ce travail est de réaliser et comparer différentes approches de compression d'images numérique.

III. 6) Conclusion :

Ce chapitre est consacré à l'évaluation de la qualité liée à la compression d'images.

Nous expliquons nos résultats en utilisant des paramètres d'évaluation de la qualité basés sur le système visuel humain (PSNR, SSIM).

Dans ce chapitre, nous décrivons un algorithme d'image (compression/décompression). Nous utilisons des images pour tester les performances de notre programme de compression et de décompression implémenté sous Matlab. Les résultats montrent que l'on peut obtenir des images reconstruites de très bonne qualité. Nous pouvons conclure que le procédé étudié permet d'obtenir de très bons résultats du point de vue de la qualité de l'image reconstruite d'une part et du taux de compression d'autre part.

CONCLUSION GENERALE

Les algorithmes de compression de données réduisent non seulement la taille des données, mais réduisent également le temps système de traitement et de transmission des données. La compression réduit la taille d'image nécessaire à l'archivage et réduit le temps nécessaire à la transmission à distance sur un réseau informatique.

L'importance des images dans notre vie quotidienne est bien connue. Les étudiants sont souvent fascinés par les images générées par ordinateur telles que les fractales. Pour les informaticiens, les formats d'image ne sont que des fichiers de données. Ces données particulières sont plus susceptibles d'être compressées par leur structure. Par conséquent, de nombreux algorithmes de compression ont été développés pour les images, certains sans perte et d'autres avec perte, ces derniers exploitants le fait que l'œil humain ne peut pas distinguer certaines nuances et détails dans une image.

Les techniques de compression avec perte fournissent généralement des taux de compression relativement élevés, mais avec un certain degré de dégradation. Par conséquent, le but des nouvelles technologies, des nouvelles techniques de compression, est de réduire davantage et de changer le moins possible.

L'objectif de notre rapport est d'obtenir une compression des données d'image tout en conservant la qualité essentielle de l'image d'origine, c'est-à-dire une bonne solution via PSNR et SSIM. Cette méthode a l'avantage d'une grande précision de reconstruction.

En pratique, chaque type de compression a ses avantages et ses inconvénients. La partie pratique de ce rapport présente les résultats que nous avons obtenus. Ces résultats sont satisfaisants, car ils démontrent une qualité d'image en termes de compression (sans/avec) perte de données. Cependant, notre méthode donne de bons résultats sur les images testées avec de très grands PSNR et SSIM.

De manière générale, on peut dire que la méthode proposée obtient de très bons résultats du point de vue de la qualité des images reconstruites.

BIBLIOGRAPHIE

- (1) REZKALLAH & YOUSFI (Compression d'images par transformée en ondelettes, UNIVERSITE DE MSILA, 2020).
- (2) OUAFI Abdelkrim (Compression d'images avec pertes par codages imbriqués, Proposition d'une optimisation de l'algorithme EZW, UNIVERSITE MOHAMED KHIDER BISKRA, 2008/2009)
- (3) ZITOUNI Athmane (Ondelettes et techniques de compression d'images numérique, UNIVERSITE MOHAMED KHIDER BISKRA, 2012/2013)
- (4) KISSOUM & ROUIFED (Watermarking et compression d'images numériques : Applications aux images médicales, UNIVERSITE MOULOUD MAMMERI, TIZI-OUZOU, 2011/2012)
- (5) BOUDJIT (Compression d'images sans perte par des techniques du codage source, Université Mohammed Seddik – Jijel, 2018/2019)
- (6) Behnas (Compression d'images fixes par bandelettes et transformée d'Alpert, Université Mohamed Khider Biskra, 2011/2012)
- (7) KOUCHKHIA (Nouvelle méthode de mesure de la qualité de compression des images basée sur une classification hiérarchique, Université Larbi ben M'hidi Oum El Bouaghi, 2012/2013)
- (8) TAOUCHE (Implémentation d'un Environnement Parallèle pour la Compression d'Images à l'aide des Fractales, Université Mentouri Constantine)
- (9) DEGHNOUCHE (ETUDE DE LA NORME DE COMPRESSION D'IMAGES JPEG, Université Mohamed Khider de Biskra, 2019)
- (10) Jouini (compression des Images dans les réseaux de capteurs multimédia sans fil, Université Larbi Tébessi, 2021)
- (11) (Image Numerique, <http://www.imagenumerique.50webs.com/index.html>, 09/03/2022, 2009/2010)
- (12) (Les Bases du Traitement d'Image, <https://patrick-bonnin.developpez.com/cours/vision/apprendre-bases-traitement-image/partie-1-introduction/>, 25/04/2022, 2016)
- (13) (Plateforme Logicielle Cépia, <https://www.pfl-cepia.inra.fr/index.php?page=tutoImg-niveaux-de-gris>, 20/03/2022, 2004/2022)
- (14) (Les différents formats d'images, <https://www.g1site.com/format-image/>, 14/03/2022, 2018)
- (15) (MATLAB, <https://www.mathworks.com/discovery/what-is-matlab.html>, 10/05/2022)