



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEURE ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE IBN KHALDOUN - TIARET

MEMOIRE

Présenté à :

FACULTÉ MATHÉMATIQUES ET INFORMATIQUE
DÉPARTEMENT D'INFORMATIQUE

Pour l'obtention du diplôme de :

MASTER

Spécialité : [Réseau et Télécommunication & Génie Informatique]

Par :

BRAHIMI Fatma

&

BELADJINE Khadidja

Sur le thème

Amélioration Automatique de la qualité visuelle des images compressées

Soutenu publiquement le / / 2019 à Tiaret devant le jury composé de :

Mr DAOUD Bachir

MAA Université Ibn Khaldoun

Président

Mr MEZZOUG Karim

MAA Université Ibn Khaldoun

Encadreur

Mr GHAFOUR Yacine

MAA Université Ibn Khaldoun

Examinateur

Remerciement

Je voudrais d'abord adresser mes vifs remerciements au Monsieur Mezzoug Karim pour son encadrement, pour son aide et ses conseils très précieux tout au long de ce travail.

Je voudrais aussi exprimer mes remerciements aux membres du jury en acceptant de juger

ce travail et d'y apporter leur caution :

❖ Mr. Daoud Bachir , département d'informatique , Université de Tiaret .

❖ Mr.Ghafour Yacine , département d'informatique, Université de Tiaret .

Que tous les enseignants qui ont contribué à ma formation trouvent en ce modeste

travail, le témoignage de ma profonde gratitude.

Je tiens aussi à remercier tous qui de près ou de loin m'ont aidé et encouragé dans ce travail.

Dédicace

Je dédie ce mémoire de fin d'étude :

Particulièrement a mon père et ma mère qui mon soutenu et


encouragé durant toute mes années d'étude

A mes frères

A mes soeurs

A tous mes amis

Ainsi qu'à tous mes camarades étudiants du département informatique

Je dédie ce travail 

Brahimi Fatma

Dédicace

*À mes chers parents, sans lesquels, je ne serai pas ici
aujourd'hui.*

À mes deux frères Ahmed et Ayoub

*À tous ceux qui m'ont soutenu et m'ont encouragé depuis mes
premiers pas dans cette vie.*

*Je leurs dédie ce modeste travail, en espérant qu'il soit à la
hauteur de leurs attentes.*

Je dédie ce travail ✍

Beladjine Khadidja

Sommaire

Remerciement	
Dédicace	
Liste des tableaux	
Liste des figures	
Résumé	
Introduction générale	1

Chapitre I : L'image Numérique

I.1.Introduction.....	3
I.2.Historique du traitement d'images	3
I.3.Introduction au traitement d'images	5
I.4.Définition de l'image	5
I.5Définition de l'image numérique	5
I.6Les caractéristiques de l'image numérique.....	6
a.Pixel	6
b.Dimension(Définition) d'une image	7
c.Résolution.....	8
g.Contraste.....	8
e.Segment	9
f.Squelette.....	9
g.Dynamique	9
h.Histogramme	9
i.Luminance	10
j.Texture.....	10
k.Contour	11
I.7.Les différents modes de représentation.....	11
a.Mode monochrome	11
b.L'image en niveaux de gris	11
c.L'image en couleurs	12
I.8.Codage des images	13

a. Avantages des images matricielles	13
b. Inconvénients des images matricielles	13
I.9. Qualité de l'image Numérique	17
I.10. Bruit et convolution	17
a. Bruit	17
b. Convolution	18
I.11. Exemples concrets de domaine d'application	18
Conclusion	20

Chapitre II

La Compression et la Décompression des Images Numériques

II.1. Introduction	22
II.2. Définition de la compression de données	22
II.2.1. Fonctionnement de la compression de données	23
II.3. Définition de la compression des images	23
II.4. Intérêt de la compression des images numériques	23
II.5. Les différents types de compression	24
II.5.1. Compression physique et compression logique	24
II.5.2. Compression symétrique et asymétrique	24
II.5.3. Compression avec ou sans pertes	25
a. Méthodes sans pertes (lossless)	25
b. Méthodes avec pertes (lossy)	25
II.6. Schéma fonctionnel de la compression des images	26
II.7. Quelques méthodes de compression	27
II.7.1. Méthodes de compression sans perte d'informations	27
a. Méthodes à base de redondances (Run Length Encoding)	27
b. Méthodes statistiques (Huffman, ...)	28
c. Méthodes à base de dictionnaires (Lempel-Ziv-Welch)	32
II.7.2. Méthodes de compression avec perte d'informations	32
II.8. Mesure de qualité de la compression d'images	32
II.9. Etude du codeur (compresseur) d'images numériques JPEG	34
II.9.1. Présentation du codec JPEG	34

II.9.2.Fonctionnement général du JPEG	35
1.Transformation de couleurs.....	35
2.Le sous-échantillonnage	36
3.Le découpage en bloc de pixels.....	37
4.Transformation DCT	37
5.La quantification	39
6.LE codage.....	39
II.9.3.Algorithme de décompression JPEG	40
II.9.4.Les avantages et inconvénients de la norme JPEG	40
Conclusion.....	41

Chapitre III

Etat de l'art sur l'amélioration visuelle des images compressées

III.1.Introduction	43
III.2.Définir le terme de qualité.....	43
a.La notion de fidélité.....	43
b.La notion de qualité	44
c.Evaluation de qualité	44
III.3.La compression des images et ses artefacts.....	45
a.L'effet de bloc.....	45
b.Le flou	46
c.L'effet d'escalier	47
III.4.Le rehaussement ou l'amélioration des images (Image Enhancement en Anglais).....	47
III.4.1.Etat de l'art des méthodes d'amélioration d'images	47
a.Des méthodes inhabituelles	48
b.Des techniques de filtrage linéaires et non linéaires	48
c.Amélioration par manipulation d'histogramme.....	49
d.Les méthodes de pré-traitements avant l'encodage.....	51
e.Les méthodes liées à l'encodage.....	52
f.Les méthodes post-traitements après le décodage de l'image	53
Conclusion.....	54

Chapitre IV

Implémentation et Tests

IV.1.Introduction	56
IV.2.L'environnement de programmation MATLAB.....	56
1.Intérêts :	57
2.Inconvénients :	57
IV.3.Image Processing Toolbox	57
IV.4.Analyse des besoins	58
IV.5.Conception & modélisation	60
a.Diagramme de classe	61
IV.6.Présentation de l'application implémentée	62
a.opérations de base	64
b.Prétraitement	64
c.Compression	65
d.Amélioration JPEG	66
e.Autres méthodes d'amélioration	67
IV.7.Résultats et discussion de l'approche proposée	68
Conclusion.....	70
Conclusion générale et perspectives	
Référence bibliographiques	

LISTE DES FIGURES

Figure I-1 Le développement du traitement d'image	4
Figure I-2 Représentation d'image numérique	6
Figure I-3 Groupe de pixels formant la lettre A	7
Figure I-4 Représentation d'une image sous la forme d'un groupe de pixels	7
Figure I-5 Représentation d'une résolution d'image (1po = 2 pixels).....	8
Figure I-6 Image avec squelette	9
Figure I-7 Exemple d'histogramme d'une image	10
Figure I-8 Image avec texture	10
Figure I-9 Image avec contour	11
Figure I-10 Image en mode monochrome.....	11
Figure I-11 Image en niveaux de gris	12
Figure I-12 Image en couleurs	13
Figure I-13 Image avec et sans bruit.....	17
Figure I-14 Domaine d'application de l'imagerie numérique	19
Figure II-1 Compression sans perte	25
Figure II-2 Compression avec perte.....	26
Figure II-3 Schéma d'un codeur source	26
Figure II-4 Principe général de la compression/décompression JPEG.....	35
Figure II-5 Transformation de couleurs du RVB au Ycbr	35
Figure III-1 L'effet de bloc	46
Figure III-2 Image avec du flou	46
Figure III-3 L'effet d'escalier	47
Figure III-4 Image avant égalisation d'histogramme.....	50
Figure III-5 Image après égalisation d'histogramme.....	50
Figure III- 6 L'étirement d'histogramme d'une image.....	50
Figure III-7 Principe de la méthode de Kutka et al : pré-traitement par rehaussement du contraste	51
Figure IV-1 l'environnement MATLAB	56
Figure IV-2 Présentation générale de l'approche proposée.....	58
Figure IV-3 Diagramme de classe	61
Figure IV-4 Page d'accueil de l'application	62

Figure IV-5 Informations sur le projet.....	62
Figure IV-6 Fermeture d'une d'interface	63
Figure IV-7 L'interface menu principale de l'application.....	63
Figure IV-8 Fenêtre d'opération de base d'une image	64
Figure IV-9 Fenêtre de Prétraitement de l'image	65
Figure IV-10 Fenêtre de compression d'image	66
Figure IV-11 Fenêtre d'amélioration d'image avec l'approche	66
Figure IV-12 Fenêtre d'amélioration d'image avec d'autres méthodes	67
Figure IV-13 Fenêtre d'amélioration de contraste.....	67
Figure IV-14 Résultat d'application des deux approches	68

LISTE DES TABLEAUX

Tableau 1 : Comparaison entre les différents formats d'images matricielles	15
Tableau 2 : Comparaison entre les différents formats d'images vectorielle.....	16
Tableau 3 : Exemples de compression sans et avec perte.....	26
Tableau 4 : Symboles avec leurs probabilités.....	29
Tableau 5 : Symboles avec leurs probabilités par ordre croissant	29
Tableau 6 : Mots de code des symboles.....	31

Résumé :

La compression d'image suscite un intérêt considérable dans le domaine de stockage et de transmission de données. Divers méthodes de compression d'images ont été élaborées dont certaines avec pertes et d'autre sans pertes. Parmi les plus utilisées et les plus efficaces on trouve la méthode de compression d'images par la DCT (Transformée en Cosinus Discrète) qui a été l'idée majeure pour la compression JPEG jusqu'à nos jours . Cette compression s'accompagne souvent d'une certaine dégradation. Le JPEG fournit des taux de compression beaucoup plus élevé, mais avec une qualité visuelle très mauvaise. Le défaut majeur de l'algorithme est dans la phase de quantification qui est irréversible. D'autres algorithmes comme le JPEG 2000 ont cherchés à résoudre ce problème en n'utilisant pas le découpage en blocs et en remplaçant la DCT par une autre transformée

Dans ce cadre s'inscrit ce projet qui porte sur le développement d'une application sous MATLAB permettant de minimiser les artefact de compression JPEG afin d'améliorer la qualité visuelle de l'image au niveau spatial.

Mots clé : image numérique, compression des images , amélioration visuelle d'images , Standard JPEG

Abstract

Image compression is of considerable interest in the field of data storage and transmission. Various image compression methods have been developed, some with loss and some without loss. Among the most used and most effective is the image compression method by the DCT (Discrete Cosine Transform) which has been the major idea for JPEG compression to the present day. This compression is often accompanied by some degradation. JPEG provides much higher compression rates, but with very poor visual quality. The major flaw of the algorithm is in the quantization phase which is irreversible. Other algorithms like JPEG 2000 have tried to solve this problem by not using the block division and replacing the DCT with another transform.

Within this framework is this project which focuses on the development of an application under MATLAB to minimize JPEG compression artefact to improve the visual quality of the image at the spatial level.

Keywords: digital image, image compression, visual image enhancement, Standard JPEG

Introduction Générale

Introduction générale

Les évolutions récentes des technologies de l'information et systèmes de communication.

ainsi que la diversification des applications et fonctionnalités du multimédia dans plusieurs domaines (la médecine, la météo, le multimédia ,le militaire, l'industrie etc.) impliquent la manipulation de données très volumineuses. Il est donc indispensable de disposer d'outils performants pour la transmission et le stockage d'énormes quantités d'informations. Pour remédier à ces contraintes, les chercheurs ont développé au cours des dernières décennies de nombreuses méthodes de compression de données déduites de la théorie de l'information et faisant appel à de nombreux domaines dont les mathématiques et l'informatique.

Parmi les données les plus volumineuses, nous comptons les images et encore les vidéo, et de ce fait ont besoin particulièrement d'un traitement adapté à leurs spécificités. Les méthodes de compression de ce type de donnés peuvent être classées suivant la nécessité de récupération parfaite ou non de l'information originale en deux grandes catégories : la compression d'information sans pertes ou réversible qui a l'avantage de préserver la qualité de l'image originale, mais avec un taux de compression relativement faible et la compression d'information avec pertes qui regroupe des algorithmes caractérisés par leur taux de compression assez élevé tout en gardant le mieux possible une qualité acceptable de l'image originale. La compression avec pertes est basée généralement sur une phase de transformation qui sert à compacter l'information utile dans un nombre minimum de coefficients non nuls.

Transformée par DCT (Transformée en Cosinus Discrète) qui est à la base de standard de compression **JPEG** qui a été adopté comme norme internationale en 1992.cette norme a beaucoup de succès et reste encore une méthode de compression d'images très performante dans le domaine de l'internet et les appareils photos numériques. Pour les hautes valeurs du taux de compression la qualité de l'image se dégrade considérablement . Ceci est dû à une étape appelée quantification, qui suit l'étape de la DCT, d'où une partie de la compression réellement se produit, les hautes fréquences sont éliminées. Ensuite, seules les basses fréquences restent et sont utilisés pour récupérer partiellement la qualité d'image. En conséquence, les images reconstruite contiennent des artefacts de qualité. ce qui limite fortement le standard JPEG.

C'est dans ce contexte qu'on propose de construire une méthode afin d'apporter une amélioration à JPEG en termes de qualité visuelle, cette méthode est basée sur deux approches d'où la première est concernée par l'élimination ou la réduction des blocs 8×8 isolés et la deuxième est concernée par le lissage des bords des blocs pixelisés.

Enfin, ce mémoire se dresse comme suit :

Le premier chapitre : est consacré à des généralités sur l'imagerie numérique

Le deuxième chapitre : donne quelques définitions de la compression, tout en passant par les deux familles d'algorithmes : réversibles (sans perte) et irréversible (avec perte). Ainsi que la chaîne de compression JPEG, tout en décrivant les différentes étapes de compression et de la décompression dans le cas avec pertes.

Le troisième chapitre : nous avons abordé l'état de l'art concernant les travaux scientifiques les plus récents pour donner aux lecteurs une vue générale sur l'actualité du domaine en question à savoir l'amélioration automatique de la qualité visuelle des images compressées.

Le quatrième chapitre : est décomposé en deux parties la première pour la description de l'environnement MATLAB plus le toolbox image processing ainsi que la méthode proposée. La deuxième partie est consacrée à quelques manipulations et expériences en utilisant l'approche proposée en plus des résultats.

Enfin, nous clôturons notre manuscrit par une conclusion générale.

Chapitre I :

L'image Numérique

Chapitre I : L'image Numérique

I.1 Introduction

L'image est un support très performant, d'où son importance et la grande quantité d'information qu'elle peut contenir, le monde s'intéresse de plus en plus et tend vers l'universalisation de son utilisation. En effet l'image a envahi plusieurs domaines de notre vie comme la médecine, la météo, le multimédia, le militaire, l'industrie etc...

L'image est l'un des moyens les plus importants qu'utilise l'homme pour communiquer avec autrui. C'est un moyen de communication universel dont la richesse du contenu permet aux êtres humains de tout âge et de toute culture de se comprendre.

Ce chapitre présente quelques concepts et notions fondamentales concernant l'image numérique tels que : la définition de l'image et de l'image numérique, les types d'image, caractéristiques d'image, codage de l'image et en fin quelques domaines d'application de l'imagerie numérique.

I.2 Historique du traitement d'images :

Au début des années 1920, le traitement de l'image numérique a vu son utilisation. Pour la transmission d'images par le câble sous-marin allant de New York à Londres. Harry G. Bartholomew et Maynard D. McFarlane effectuent la première numérisation d'image avec compression de données pour envoyer des fax de Londres à New York. Le temps de transfert passe ainsi de plus d'une semaine à moins de trois heures. Il n'y a pas vraiment d'évolution par la suite jusqu'à la période d'après-guerre.

Le traitement du signal prend de l'importance vers la fin de la Seconde Guerre mondiale avec l'arrivée du radar. La prospection pétrolière participe aussi beaucoup au développement des techniques du traitement de signal.

Le véritable essor du traitement d'images n'a lieu que dans les années 1960 quand les ordinateurs numériques de troisième génération ont commencé à offrir la vitesse et les capacités de stockage requises pour la mise en œuvre pratique des algorithmes de traitement d'image. Peu après, la redécouverte de la transformée de Fourier rapide (FFT) révolutionne le domaine, en rendant possible les manipulations du contenu fréquentiel des signaux sur ordinateur.

Cependant, l'essentiel des recherches porte encore, à cette époque, sur l'amélioration des images et leur compression.

En 1980, David Marr formalise le premier la détection de contours de manière précise (D.Marr et E. Hildreth:*Theory of Edge Detection*, Proc. R. Soc. London, B 207, 187-217, 1980).

Au cours des années 1980, un véritable engouement se fait jour pour le traitement de l'image et surtout pour la compréhension de l'image par des systèmes experts. Les ambitions étaient beaucoup trop grandes, l'échec fut d'autant plus cuisant.

Les années 1990 sont témoin de l'amélioration constante des opérateurs. La recherche médicale devient un très gros demandeur en traitement d'images pour améliorer .

Les diagnostics faits à partir des nombreuses techniques d'imagerie médicale, la technique reine étant l'IRM. Les publicitaires, puis le grand public se familiarisent avec la retouche d'image grâce au logiciel Photoshop, et le traitement d'images dans un objectif esthétique se répand avec l'apparition d'autres logiciels dédiés (The Gimp, Paint Shop Pro). Enfin, la décennie s'achève sur l'engouement pour les ondelettes et les images multimodales.[1]



Figure I-1 Le développement du traitement d'image

I.3 Introduction au traitement d'images :

Le traitement d'images est un domaine très vaste qui a connu, et qui connaît encore, un développement important depuis quelques dizaines d'années. [1]

Le traitement d'images désigne une discipline de l'informatique et des mathématiques appliquées qui est utilisée dans plusieurs domaines (Imagerie aérienne et spatiale, Technologies biomédicales, Robotique, Télé surveillance, Physique nucléaire.....etc).

Il s'agit d'un ensemble des méthodes et techniques opérant sur l'image numérique afin de rendre cette opération possible, plus simple, plus efficace et plus agréable, d'améliorer l'aspect visuel de l'image et d'en extraire des informations jugées pertinentes.

I.4 Définition de l'image :

Une image est une représentation planaire d'une scène ou d'un objet situé en général dans un espace tridimensionnel.[w1] Son élaboration résulte de la volonté de proposer une entité observable par l'œil humain. Ceci explique d'une part son aspect planaire et d'autre part le fait que l'information élémentaire associée à chaque point de l'image soit transcrite en niveau de gris ou en couleur.

I.5 Définition de l'image numérique :

Le terme d'image numérique désigne, dans son sens le plus général, toute image qui a été **acquise**, **traitée** et **sauvegardée** [w2] sous une forme codée représentable par une série de bits.

Acquise : par des dispositifs à convertisseur analogique-numérique.

Traité : grâce à des outils graphiques.

Sauvegardée : sur un support informatique.

L'image numérique est l'image dont la surface est divisée en éléments de tailles fixes appelés cellules ou pixels, ayant chacun comme caractéristique un niveau de gris ou de couleurs prélevé à l'emplacement correspondant dans l'image réelle, ou calculé à partir d'une description interne de la scène à représenter. [2]

La numérisation est la transformation d'un signal analogique en un signal numérique. elle comporte deux activités parallèles : **l'échantillonnage** et **la quantification**.

➤ **L'échantillonnage** : consiste la discrétisation de coordonnées de l'image réelle. Son résultat est un plan à points discrets, ces points constituent l'ensemble des éléments composant l'image numérique

➤ **La quantification** : consiste la discrétisation d'intensités de l'image réelle. Son résultat est un intervalle de valeurs entières qui peut prendre un pixel. Ces valeurs définissent la nuance des niveaux de gris ou bien les niveaux de couleurs qui peuvent être prises par un pixel de l'image.

Elle est alors la conversion d'une distribution continue d'intensités lumineuses dans un plan xOy en des valeurs numériques $f(x,y)$, représentée par une matrice bidimensionnelle comme la montre la figure I.2 où x, y : coordonnées cartésiennes d'un point de l'image.

$F(x, y)$: niveau d'intensité.

La valeur en chaque point exprime la mesure d'intensité lumineuse perçue par le capteur.

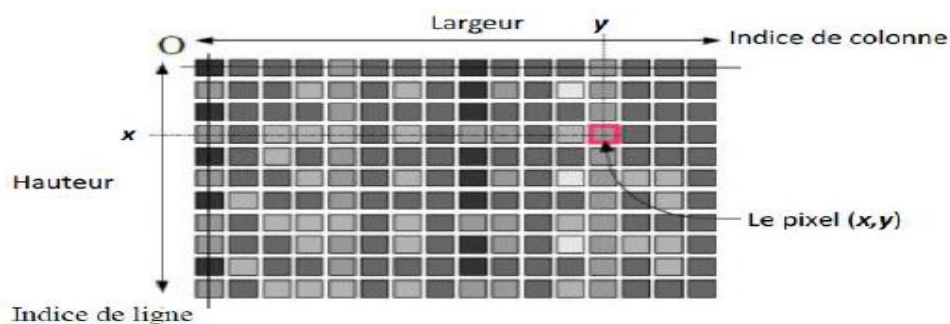


Figure I-2 Représentation d'image numérique

I.6 Les caractéristiques de l'image numérique :

L'image est un ensemble structuré d'informations caractérisé par les paramètres suivants :

A. Pixel

Une image est constituée d'un ensemble de points appelés pixels (pixel est une abréviation de PICture ELe ment). Le pixel représente ainsi le plus petit élément constitutif d'une image numérique. L'ensemble de ces pixels est contenu dans un tableau à deux dimensions constituant l'image. [3]

Pour faire simple, un pixel est un minuscule carré de couleurs qui, une fois associé et arrangé en lignes et colonnes sur votre écran, formera votre image. la lettre A, par exemple, peut être affichée comme un groupe de pixels dans la figure ci-dessous.

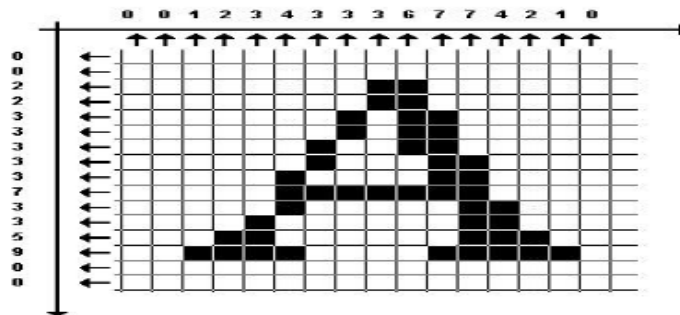


Figure I-3 Groupe de pixels formant la lettre A

B. Dimension(Définition) d’une image :

On appelle définition le nombre de points (pixels) constituant une image. Le nombre de lignes de cette matrice multiplié par le nombre de colonnes nous donne le nombre total de pixels dans une image. La figure ci-dessous montre un sous-tableau de 5×5 pixels [4]

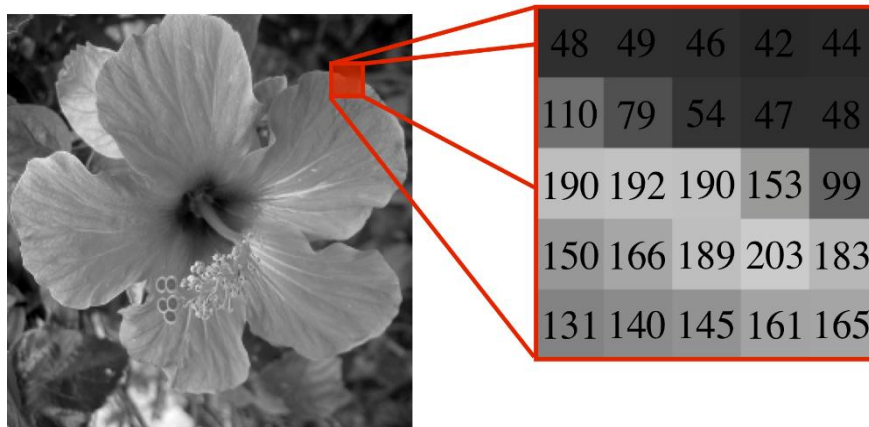


Figure I-4 Représentation d’une image sous la forme d’un groupe de pixels

C. Résolution

Terme souvent confondu avec la "définition", détermine par contre le nombre de pixels par unité de longueur, exprimé en points par pouce (PPP, en anglais DPI Dots Per Inch).[3]

La résolution permet ainsi d'établir le rapport entre le nombre de pixels d'une image et la taille réelle de sa représentation sur un support physique, plus la résolution baisse, plus le nombre de pixels dans l'image diminue, et plus la qualité de l'image numérique se dégrade donc il est conseillé de tjrs choisir une résolution qui soit suffisamment élevé pour obtenir une image de bonne qualité.

On peut distinguer deux modes de résolution des images :

- Résolution spatiale : est le plus petit détail discernable.
- Résolution tonale : est le plus petit changement discernable.

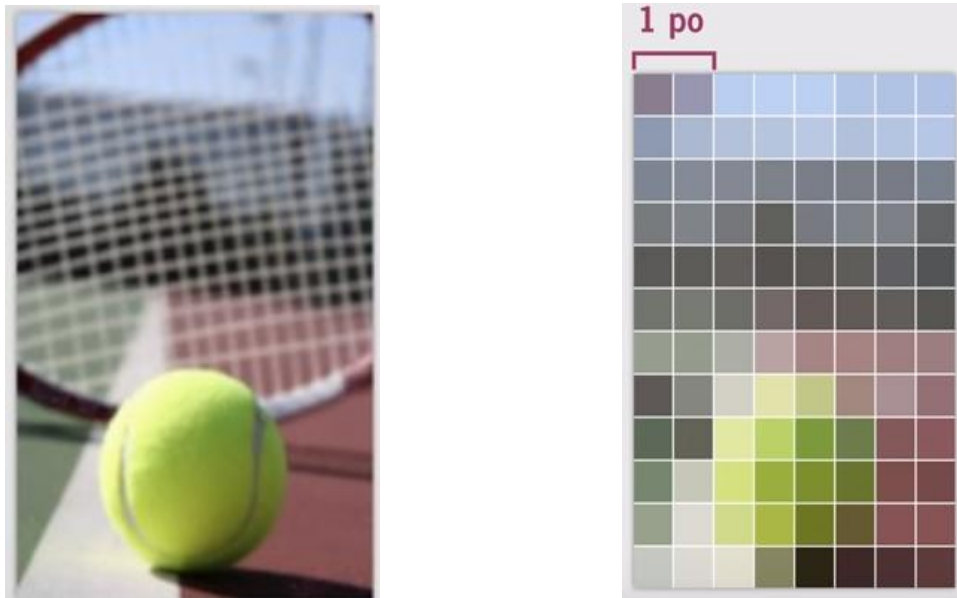


Figure I-5 Représentation d'une résolution d'image (1 po = 2 pixels)

D. 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. Le contraste peut être défini de plusieurs façon :[5]

- Ecart type des variations des niveaux de gris

$$(I.1) \quad C = \sqrt{\frac{1}{M \times N} \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} (f(x,y) - Moy)^2}$$

E. Segment

Une portion de droite limitée par deux points. Ces deux points sont les extrémités du segment. Ainsi, le segment [AB] a pour extrémités les points A et B.

F. Squelette

Est une représentation filaire des objets dans une image. Qui doit garantir les propriétés de préservation de la topologie ainsi que la géométrie de l'objet à traiter. D'autres propriétés du squelette telles que la restructurabilité et l'épaisseur unitaire sont requises dans certains cas d'utilisation. Quelques exemples de squelettes sont montrés sur la figure suivante :

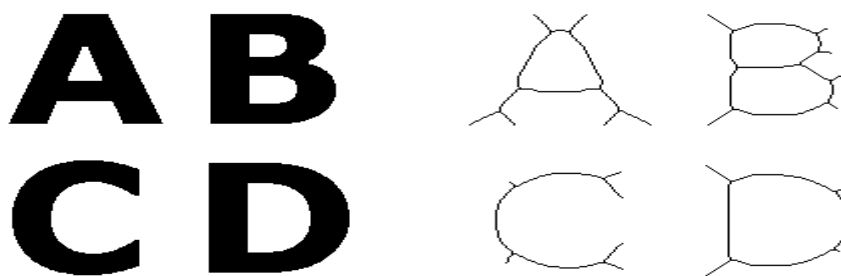


Figure I-6 Image avec squelette

G. Dynamique

La dynamique d'une image, c'est la capacité d'une image numérique à redonner les vraies échelles d'intensité lumineuses du monde réel. On exprime souvent la dynamique en utilisant l'unité des stops. [w3] Cela se calcule très simplement en faisant le rapport de la valeur la plus claire mesurable sur la valeur la plus sombre mesurable et en prenant le logarithme népérien

Exemple : fichier JPEG

Pixel le plus clair : 255

Pixel le plus sombre : 0 \Rightarrow 256 valeurs possibles $\text{stops} = \log(256) / \log(2) = 8$

H. Histogramme

Un histogramme est la représentation graphique des pixels qui composent une image. Ces pixels sont répartis en fonction de leur luminosité. Il permet de donner un grand nombre d'informations sur la distribution des niveaux de gris ou de couleur et il peut être

utilisé pour améliorer la qualité d'une image en introduisant quelques modifications pour pouvoir extraire les informations utiles de celle-ci. [3]

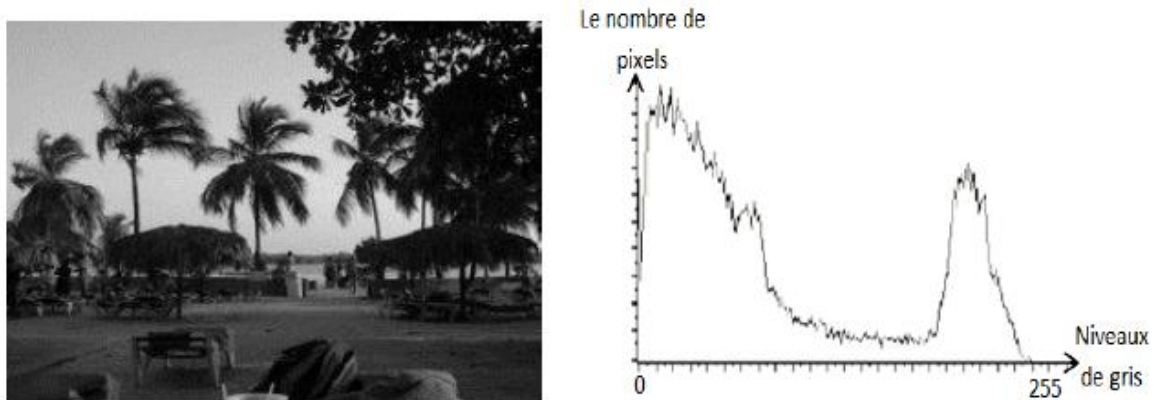


Figure I-7 Exemple d'histogramme d'une image

I. Luminance

C'est le degré de luminosité des points de l'image. Elle est définie aussi comme étant le quotient de l'intensité lumineuse d'une surface par l'aire apparente de cette surface, pour un observateur lointain, le mot luminance est substitué au mot brillance, qui correspond à l'éclat d'un objet. Une bonne luminance se caractérise par : [6]

- Des images lumineuses (brillantes).
- Un bon contraste : il faut éviter les images où la gamme de contraste tend vers le blanc ou le noir; ces images entraînent des pertes de détails dans les zones sombres ou lumineuses.
- L'absence de parasites.

J. Texture

répétition spatiale d'un même motif dans différentes directions de l'espace. Cette définition est limitative car elle caractérise l'objet indépendamment d'un observateur humain. La notion de texture est utilisée pour traduire un aspect homogène de la surface d'un objet sur une image.[8]



Figure I-8 Image avec texture

K. Contour

Parties les plus informatives d'une image qui permet de repérer les différents objets qui constituent la scène de l'image.[8]



Figure I-9 Image avec contour

I.7. Les différents modes de représentation

Il existe différentes catégories d'image selon le nombre de bit sur lequel est codée la valeur de chaque pixel.

A. Mode monochrome

Le mode monochrome est le plus simple chaque pixel est soit allumé « Blanc », soit éteint « Noir », l'image obtenue n'est pas très nuancée (figure I.10). Alors, pour convertir une image couleur en mode monochrome il faut d'abord passer par le mode niveaux de gris.



Figure I-10 Image en mode monochrome

B. L'image en niveaux de gris

Le niveau de gris est la valeur de l'intensité lumineuse en un point. La couleur du pixel peut prendre des valeurs allant du noir au blanc en passant par un nombre fini de niveaux intermédiaires. Donc pour représenter les images à niveaux de gris (figure I.11).

on peut attribuer à chaque pixel de l'image une valeur correspondante à la quantité de lumière renvoyée. Cette valeur peut être comprise par exemple entre 0 et 255. Chaque pixel n'est donc plus représenté par un bit, mais par un octet [6]



Figure I-11 Image en niveaux de gris

C. L'image en couleurs

Même s'il est parfois utile de pouvoir représenter des images en noir et blanc ou en niveau de gris, les applications multimédias utilisent le plus souvent des images en couleurs (**figure I.12**). La représentation des couleurs s'effectue de la même manière que les images monochromes avec cependant quelques particularités. En effet, il faut tout d'abord choisir un modèle de représentation.[6]

Pour cela on utilise un espace de couleur à plusieurs dimensions qui consiste à donner suffisamment de composantes numériques pour décrire une couleur. Il y a des différentes représentations des images couleur :

- La représentation en couleurs réelle sur 24 bits.
- La représentation en couleurs indexées ; on utilise une table appelée palette pour éviter la redondance de couleur.
- Le mode RGB est idéal pour l'affichage sur écran, une image RGB 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 toute nuance de couleur.
- Le mode RGB correspond à l'affichage des moniteurs, où chaque point affiché est décomposé d'un mélange de lumière RGB.



Figure I-12 Image en couleurs

I.8 Codage des images

Il y a deux modes de codage d'une image numérique :

- Matricielle
- Vectoriel

Images matricielles

Une image matricielle est constituée de pixels et se réduit donc à une matrice de points. (Images pixellisées), chacun de ces points possédant une ou plusieurs valeurs décrivant sa couleur. [7]

A. Avantages des images matricielles :

- Les images matricielles peuvent facilement être créées et stockées dans un tableau de pixels représentant l'image.
- Adaptable aux images complexes.
- Les images matricielles peuvent facilement être affichées sur un écran ou être imprimées.

B. Inconvénients des images matricielles :

- Les fichiers peuvent être trop gros (nécessite de compression).
- Les dimensions de l'image doivent être prévues pour la résolution de l'interface de sortie (écran, imprimante).
- Lorsqu'on agrandi trop une image matricielle, on perd la qualité « pixellisation ».

Les formats de l'image matricielle :

Il existe beaucoup de formats de fichiers pour sauvegarder les images matricielles : TIFF, GIF, JPEG, PNG, BMP,...

- **Le format GIF :**

Le format GIF (**Graphic Interchange Format**) est un format léger qui peut également contenir des animations. Une image GIF ne peut contenir que 2, 4, 8, 16, 32, 64, 128 ou 256 couleurs parmi 16.8 millions dans sa palette en mode RVB. Elle supporte également une couleur de transparence.

- **Le format TIFF :**

TIFF (Tagged Image Format File) spécifié par Microsoft Corporation et Aldus Corporation, format multi-usage avec/sans compression.

- **Le format JPEG :**

Le format JPEG ou JPG (Joint Photographic Expert Group), très couramment utilisé pour le codage des images bitmap et des photos, est un format de compression très efficace basé sur la méthode DCT (Discret Cosin Transform).

- **Le format BMP :**

Le format BMP (BitMaP) est un format d'image matricielle ouvert développé par Microsoft et IBM. C'est l'un des formats d'images les plus simples à développer et à utiliser pour programmer. Il est lisible par quasiment tous les visualiseurs et éditeurs d'images.

Ci-dessous un tableau comparatif entre les différents formats d'images matricielles :

Nom du format	Points forts	Points faibles	Notes
JPEG JPEG 2000 (Joint Photographic Experts Group)	Compression Excellente	Compression Destructrice	Spécialement conçu pour les photographies, il est cependant à utiliser avec délicatesse tant sa compression peut brouiller l'image. Le format JPEG 2000, évolution du format original, peut être réglé pour compresser sans pertes.
GIF (Graphical Interchange Format)	Possibilité d'animation et de transparence, compression efficace	Limité à 256 couleurs	Très Répandu sur le Web malgré ses faiblesses et un problème de droit sur son format de compression. À déconseiller pour les photos.
TIFF			Format de stockage très

(Tagged Image File Format)	Compression sans perte efficace. Couche de transparence.	Lourdeur des fichiers non compressés. Format propriétaire.	utilisé, à éviter pour le Web.
BMP (Bitmap)	Format par défaut de Windows Qualité originale	Disponible uniquement sur la plateforme de Microsoft.	Généralement non compressé et de ce fait des fichiers très « Lourds »

Tableau 1 : Comparaison entre les différents formats d'images matricielles

Images vectorielles

Les images vectorielles sont des représentations d'entités géométriques telles qu'un cercle, un rectangle ou un segment. Ceux-ci sont représentés par des formules mathématiques (un rectangle est défini par deux points, un cercle par un centre et un rayon, une courbe par plusieurs points et une équation). C'est le processeur qui sera chargé de "traduire" ces formes en informations interprétables par la carte graphique.

A. Avantage des images vectorielles :

- L'image vectorielle est constituée uniquement d'entités mathématiques, il est possible de lui appliquer facilement des transformations géométriques (zoom, étirement, ...), tandis qu'une image bitmap, faite de pixels, ne pourra subir de telles transformations qu'au prix d'une perte d'information
- Peuvent aisément être redimensionnées.
- Prennent moins de place qu'une image bitmap.[7]

B. Inconvénients des images vectorielles :

- Chaque format de fichier vectoriel possédant ses propres attributs, la compatibilité entre les formats est difficile
- les fichiers des images vectorielles sont souvent fragiles (c'est-à-dire que la perte d'informations due à des fausses manipulations peut être irréversible, et cela plus fréquemment qu'avec des images bitmap
- Peuvent difficilement stocker des images complexes comme des photographies.

C. Les formats de l'image vectorielle :

- AI (Adobe Illustrator)
- PICT (Picture).

- EPS/PS(Encapsulated PostScript /PostScript).
- PDF(Portable Document Format).

Le tableau au-dessous illustre la différence entre les formats vectoriels de compression :

Nom du format	Points forts	Points faibles	Notes
AI (Adobe Illustrator)	Reconnu par tous les logiciels graphiques.	Format propriétaire	Format standard de Adobe Illustrator, l'un des plus utilisés du fait de la popularité du logiciel.
EPS/PS (Postscript/ Encapsulated Postscript)	Très bien reconnu sur tous les systèmes.	N'a d'intérêt que dans le cadre d'une impression. Fichier très lourd.	Format hybride bitmap/vectoriel, réservé à l'impression. EPS est un fichier PS qui comporte quelques restrictions supplémentaires.
PDF (Portable Document Format)	Affiche les documents.	Taille prohibitive. Ne peut se lire qu'avec le logiciel Acrobat ou logiciel équivalent.	Version simplifiée de PostScript, il a été conçu pour afficher les documents de la même manière quel que soit le système.
PICT (Picture)	Format par défaut de Mac OS, donc encore utilisé.	Disponible uniquement sur la plateforme d'Apple.	N'a plus grand intérêt face aux autres formats existants.

Tableau 2 : Comparaison entre les différents formats d'images vectorielle

I.9. Qualité de l'image Numérique

Elle dépend d'une part, de la qualité des images d'origine, et d'autre part, des moyens mis en œuvre pour convertir un signal analogique en signal numérique. Elle dépend aussi de plusieurs facteurs : [3]

- La qualité des périphériques de numérisation de l'image (convertisseur analogique-numérique), du nombre de niveaux de gris ou de couleurs enregistrées.
- La qualité de l'affichage à l'écran : définition et résolution de l'écran.
- Les prétraitements d'images.

I.10. Bruit et convolution

A. Bruit

Le bruit est une information parasite qui s'ajoute aléatoirement à une image numérique causant la perte de la netteté dans les détails. ce bruit peut être d'origines diverses :

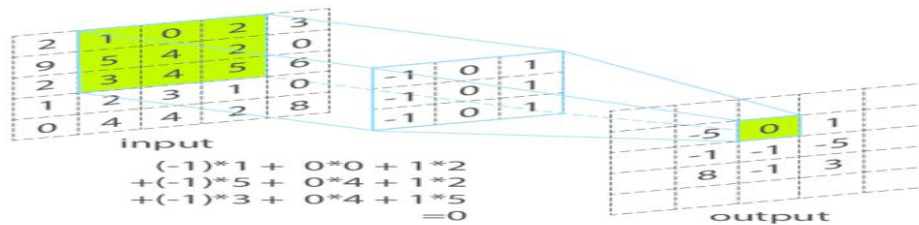
- Bruits d'acquisition : flou, poussière, bougé
- Bruits lié au système d'acquisition (capteur) : mauvaise mise au point, bruit thermique, bruit de grenaille.
- Bruits d'échantillonnage et de quantification
- Bruits lié à la transmission



Figure I-13 Image avec et sans bruit

B. Convolution

La convolution est le remplacement de la valeur d'un pixel par une combinaison de ses avoisinants. Elle consiste à faire balayer une fenêtre d'analyse (noyau de convolution ou masque de convolution.) sur l'ensemble des pixels de l'image.



L'opération de convolution est calculée en tout point de l'image source en 3 étapes :

- On centre le masque (par exemple un masque carré 3*3) sur le pixel courant.
- On calcule les 9 produits entre la valeur de l'image et la valeur du masque superposé.
- Puis on somme les 9 produits pour obtenir la valeur du pixel de l'image filtrée. [w3]

I.11.Exemples concrets de domaine d'application

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 :

- le domaine militaire :
 - missiles en tous genres (autodirecteur (courte portée), de croisière (longue portée), etc.),
 - le renseignement (télédétection à partir d'images satellite dont la précision peut aller aujourd'hui jusqu'à quelques centimètres, ou aériennes, la photo-interprétation),
 - les simulateurs « réels » (de vol avion, de char, etc.), [w3]
- le domaine médical :
 - angiographie,
 - radiographie,
 - échographie,
 - scanner,
 - IRM, etc.

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

- du domaine industriel :
 - le contrôle de la qualité des produits en bout de chaîne (état de surface, dimensionnement, forme, couleur, présence des comprimés par exemple dans l'industrie pharmaceutique),
 - le contrôle non destructif,
 - compris dans l'industrie alimentaire en plus du contrôle de la composition, de la fraîcheur, etc.

Jusqu'au domaine du multimédia :

- la compression / décompression pour la transmission d'Images ou le stockage,
- l'amélioration de la qualité,
- recherche d'images par le contenu dans des banques d'images,
- la TV haute définition (Standard MPEG4),
- le coloriage des dessins animés, des vieux films N&B,
- le remplacement des panneaux publicitaires dans des retransmissions sportives,
- le suivi optimal automatique du porteur de balle dans un jeu tel que le football américain,
- le tatouage d'image (pour prouver l'authenticité des documents, ainsi que leur appartenance),

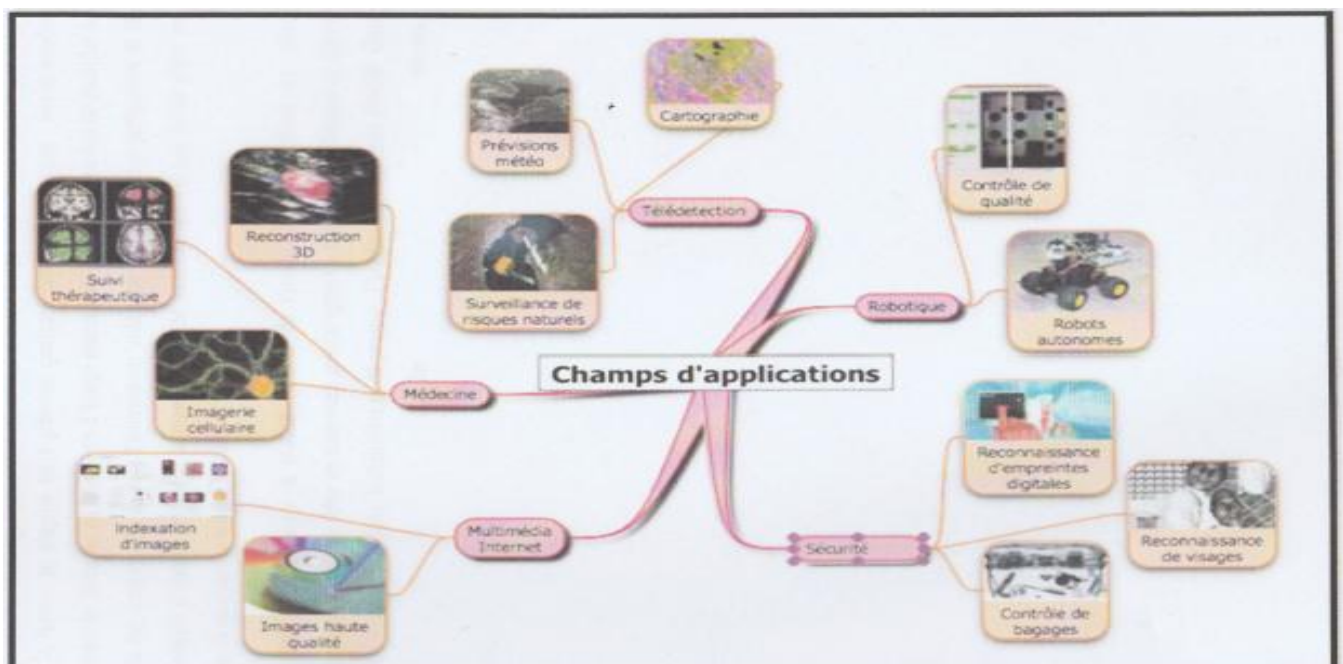


Figure I-14 Domaine d'application de l'imagerie numérique

Conclusion

Dans ce chapitre, nous avons essayé de faire un récapitulatif sur les notions Élémentaires d'image numérique (vectorielle et matricielle) comme pixel, dimension, Résolution, bruit, histogramme, contours et textures, luminance, contraste, niveau de gris et la couleur et les différents formats (Bitmap, JPEG, PNG, TIFF, GIF.).Le chapitre qui suit est consacré aux méthodes et les différentes techniques de compression et de décompression.

Chapitre II

La Compression et la Décompression des Images
Numériques

Chapitre II La Compression et la Décompression des Images Numériques

II.1 Introduction

De nos jours, grâce au développement de l'outil informatique, aux réseaux et à Internet particulièrement la quantité d'information échangée entre les usagers a considérablement augmenté. Ceci nécessite des supports de stockage puissants et performants pour satisfaire la forte demande de l'espace de stockage, et garantir un minimum de trafic sur un réseau qui souffre de bande passante limitée. Des études ont été menées afin de mettre en évidence des algorithmes de compression et de décompression afin de réduire la taille des données en exploitant la puissance des processeurs plutôt qu'en augmentant les capacités de stockage et de transmission des données. Nous présentons dans ce chapitre les principales méthodes de codages et de compression d'image numérique, nous décrirons aussi l'algorithme de base de la norme JPEG.

II.2 Définition de la compression de données

La compression de données d'une manière général est l'ensemble des méthodes consistant à réduire un volume important de données sans se préoccuper de leurs origines (images, textes, son...). Elle permet donc de diminuer la taille de stockage et de rendre possible leur transports sur des réseaux de communication (internet, GSM, câble, TV satellite).[10]

Vu les variétés des données (variétés des domaines d'utilisations des informations), de nombreux algorithmes de compression existent, chacun ayant sa particularité et surtout un type de données cible. Car toutes les données ne se compressent pas de la même manière. Un algorithme de compression de texte travaillera sur les répétitions du nombre de caractères ou de parties de phrases. Un algorithme de compression d'images travaillera quant à lui sur d'autres domaines comme la différence entre un pixel et un autre. On imagine cependant mal le second algorithme en train de compresser un texte. par exemple, un algorithme qui donne de bons résultats pour la compression de fichiers textes, peut aboutir à de mauvais résultats pour la compression d'images [10]. Néanmoins, tous ces algorithmes ont un point commun : leur objectif est de récupérer les données initiales (partiellement voire totalement) [w1]

II.2.1 Fonctionnement de la compression de données

La compression de données suit deux étapes toujours présentes dans le processus :

- la compression
- la décompression.

La première consiste à appliquer l'algorithme de compression pour réduire la taille des données. On obtient dès lors un fichier dît compressé. Cette étape est effectuée avec un outil appelé compresseur.

La seconde consiste à rétablir les données d'origine à l'aide du fichier compressé. Elle consiste souvent à appliquer l'algorithme de compression en sens inverse. Cette étape est effectuée avec un outil appelé décompresseur. [w1]

II.3 Définition de la compression des images

La compression d'image est une application de la compression de données sur des images numériques [w2]. Cette compression consiste à réduire la quantité d'information d'une image afin de pouvoir l'emmagasiner sans occuper beaucoup d'espace et la transmettre rapidement à travers les réseaux informatiques. Pour cela Il faut utiliser des algorithmes de compression qui ne dégradent pas la qualité des images, en particulier lorsque celles-ci présentent un niveau de détail important (schéma, document technique, peinture, ...). [w4]

II.4 Intérêt de la compression des images numériques

Habituellement, lorsque l'on pose la question « Pourquoi compresse-t-on les images ? », tout le monde qu'il soit informaticien ou non répondra par la même réponse : « Pour gagner l'espace de stockage ». Bien évidemment, cette réponse est correcte et elle était probablement la seule raison qui poussa initialement à la compression. Mais aujourd'hui, d'autres utilisations ont été trouvées et cette partie va donc vous les présenter.

- **Economie d'espace disque /stockage** : une seule image ne changera pas beaucoup de chose, mais imaginez-vous devoir stocker des centaines voire des milliers d'images ... Le stockage sera tôt ou tard un problème. [w3]

- **Expérience utilisateur sur le web** : quand on est Webmaster, on doit attacher beaucoup d'importance à l'expérience utilisateur sur son site web. Si les pages sont longues à charger, les visiteurs risquent de fuir.

- **Référencement Naturel** : La rapidité d'un site internet est l'une des critères / algorithmes SEO de Google. Compresser ses images participe donc à l'optimisation SEO d'un site (pour search engine optimization), (est un ensemble de techniques visant à optimiser la visibilité d'une page web dans les résultats de recherche d'un site).

- **La sécurité** : une dernière raison à la compression est la sécurité tout d'abord, il faut savoir que la compression est très utilisée dans le cryptage, puisque les réseaux numériques de communication sont largement utilisés pour l'échange des informations alors la sécurité de ces informations échangées est devenue une nécessité primordiale dans beaucoup d'applications des organismes civils ou militaires. Au-delà de ça, il existe une protection applicable uniquement par compression afin d'assurer la confidentialité et d'empêcher toute modification ou exploitation non autorisée des données. L'une des méthodes connues pour la réalisation efficace de cet objectif est la compression qui rend complètement les données illisible et imperceptible.

II.5 Les différents types de compression

On distingue plusieurs types de compression telle que :

II.5.1 Compression physique et compression logique

Une distinction doit cependant être faite entre la compression physique et la compression logique :

La compression physique : s'applique uniquement aux données de l'image. Il s'agit de translater les trains de bit d'un motif à un autre.

La compression logique : est effectuée par un raisonnement logique en substituant une information par une information équivalente.

II.5.2 Compression symétrique et asymétrique

Une méthode de compression symétrique utilise le même algorithme, et demande la même capacité de calcul, aussi bien pour la compression que pour la décompression.

Par exemple : une application de transmission de données où la compression et la décompression sont les deux faits en temps réel sera généralement implémenté avec un algorithme symétrique si l'on veut atteindre la plus grande efficacité.

Par contre les méthodes de compression asymétriques demandent plus de travail pour l'une des deux opérations. La plupart des algorithmes requiert plus de temps de traitement

pour la compression que pour la décompression, par exemple lorsque l'on archive des données auxquelles on accède peu souvent pour des raisons de sécurité.

II.5.3 Compression avec ou sans pertes

On peut distinguer deux grandes familles d'algorithmes de compression : les méthodes dites sans perte ou réversibles garantissent la restitution parfaite des images, alors que les méthodes dites avec perte ou irréversibles modifient plus ou moins la valeur des pixels [2]

A. Méthodes sans pertes (lossless) :

Cela signifie que lorsque des données sont compressées et ensuite décompressées, l'information originale contenue dans les données a été préservée. Aucune donnée n'a été perdue ou oubliée. Les données n'ont pas été modifiées. [3]

Cette compression conservatrice est utilisée dans des applications comme l'archivage des images médicales, l'imagerie satellitaire là où le coût des images est élevé et les détails sont importants, les textes, les programmes et tout autre type de donnée nécessitant une conservation à l'identique des données. Le seul critère d'évaluation des performances est dans ce cas "Le taux de compression".

La figure 1 illustre schématiquement la compression sans perte :

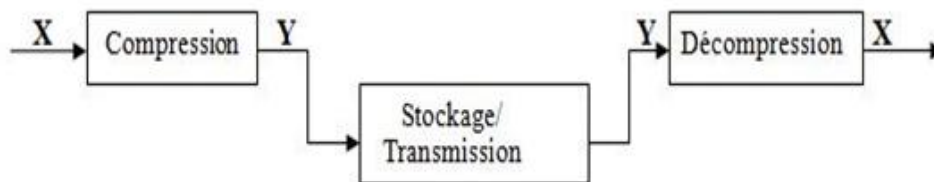


Figure II-1 Compression sans perte

B. Méthodes avec pertes (lossy) :

Cette compression a un rôle tout à fait différent par rapport à celle de sans perte. Le principe de la compression avec perte est d'avoir une permission de supprimer quelques données (qui sont inutiles) de l'information pour avoir une meilleure compression, ce qui signifie qu'après une décompression du compressé on aura un résultat approximatif du fichier original. On utilise souvent cette compression pour compresser les données comme image, son, vidéo, car pour ce type de données on peut enlever quelques données sans toucher aux informations essentielles qui sont contenues dans ce type de fichier.[4]

Voici un schéma représentatif d'une compression avec perte :

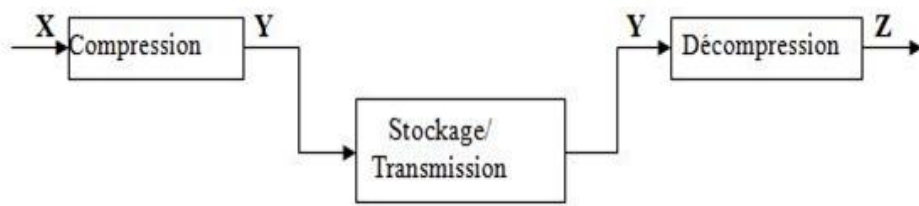


Figure II-2 Compression avec perte

Le tableau suivant illustre quelques exemples de méthodes de compression :

Méthodes de compression	Exemples
Sans Perte	<ul style="list-style-type: none"> • Codes à longueur variable : Shannon-Fano, Huffman • A Base de dictionnaire : Lempel-Ziv (LZ77, LZW) • Par Décorrélacion : Run-Length Encoding (RLE) • Codage arithmétique
Avec Perte	<ul style="list-style-type: none"> • Par transformée en cosinus discrète (DCT) : JPEG • Par transformée en ondelettes discrète (DWT) : JPEG 2000 • Par moyennage de blocs • Par quantification (scalaire ou vectorielle)

Tableau 3 Exemples de compression sans et avec perte

II.6 Schéma fonctionnel de la compression des images

Tout système de compression peut se décomposer en trois modules distincts : la transformation des données images source, la quantification et l'affectation de codes binaires. [6] .la compression d'une image se fait en général selon le schéma fonctionnel de la présenté dans la figure II.3 ci-dessous :

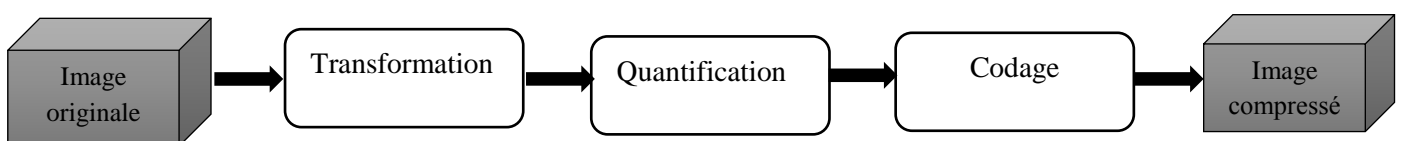


Figure II-3 Schéma d'un codeur source

1) Transformation :

Pour réduire le volume d'information on effectue une opération de décorrélation des pixels (La dépendance existante entre chacun des pixels et ses voisins traduit une corrélation très forte sur l'image. On essaie donc de tirer parti de cette corrélation) La décorrélation consiste à transformer les pixels initiaux en un ensemble de coefficients moins corrélés, c'est une opération réversible.

2) Quantification :

La quantification des coefficients a pour but de réduire le nombre de bits nécessaires pour leurs représentations. Elle représente une étape clé de la compression. Elle approxime chaque valeur d'un signal par un multiple entier d'une quantité q , appelée quantum élémentaire ou pas de quantification.

3) Codage :

Le codage entropique effectue un codage sans perte sur les valeurs quantifiées. Cette dernière étape est nécessaire dans les méthodes sans perte, mais elle est souvent présente aussi dans les algorithmes irréversibles, puisque les valeurs transformées et quantifiées contiennent davantage de redondances [2]. Un codeur doit satisfaire à priori les conditions suivantes :

- Unicité : deux messages différents ne doivent pas être codés de la même façon.
- Déchiffabilité : deux mots de codes successifs doivent être distingués sans ambiguïté.

II.7 Quelques méthodes de compression

La plupart des méthodes de compression visent à enlever la redondance présente dans l'image de manière à diminuer le nombre de bits nécessaires à sa représentation. Les techniques de compression les plus élémentaires sont décrites ici :

II.7.1 Méthodes de compression sans perte d'informations

Parmi ces méthodes, on trouve : la méthode RLE, la méthode HUFFMAN...etc.

A. Méthodes à base de redondances (Run Length Encoding)

C'est la méthode la plus simple et la plus utilisée. Son principe de base consiste à rechercher des données redondantes. Elle est utilisée par de nombreux formats d'images (BMP, PCX, TIF).

Algorithme de compression

1. Recherche des caractères répétés plus de n fois (n fixé par l'utilisateur)
2. Remplacement de l'itération de caractères par :
 - un caractère spécial identifiant une compression.
 - le nombre de fois où le caractère est répété.
 - le caractère répété.

Algorithme de décompression

Durant la lecture du fichier compressé, lorsque le caractère spécial est reconnu, on effectue l'opération inverse de la compression tout en supprimant ce caractère spécial [10].

Exemple :

AAAAARRRRRROLLLLBBBBBUUTTTTT

On choisit comme caractère spécial : @ et comme seuil de répétition : 3

Après compression : @5A@6RO@4L@5BUU@6T gain : 11 caractères soit 38%.

Utilité : essentiellement pour la compression des images (car une image est composée de répétitions de pixels, de couleur identique, codés chacun par un caractère).

Problème : Inefficace sur des images très texturées

B. Méthodes statistiques (Huffman, ...)

Le codage Huffman [8], crée des codes à longueurs variables sur un nombre entier de bits.

L'algorithme considère chaque message à coder comme étant une feuille d'un arbre qui reste à construire. L'idée est d'attribuer aux messages de plus faibles probabilités, les mots codés les plus longs et les mots codés les plus courts pour les messages de fortes probabilités, il est utilisé par les formats d'images JPEG.

Principe :

repose sur la création d'une structure d'arbre composée de nœuds. la méthode est la suivante :

1. Calculer la probabilité associée à chaque symbole dans la chaîne à coder.
2. Classer les probabilités d'apparition des symboles par ordre croissants.
3. Sommer les deux plus faibles probabilités correspondantes aux deux symboles de la séquence (la plus petite probabilité prend un 0 et celle plus grande prend un 1) ; ce qui nous donne une nouvelle probabilité.

4. Répéter les opérations 2 et 3, jusqu'à ce qu'il ne reste que deux probabilités (la racine).

5. Lire le code binaire de chaque symbole de haut vers le bas.

Exemple de codage Huffman :

Soit une source composée d'un alphabet de 8 symboles avec les probabilités données sur le tableau II.1 :

Symboles	C	E	B	D	F	A
Probabilité	0.25	0.35	0.10	0.15	0.05	0.10

Tableau 4 Symboles avec leurs probabilités.

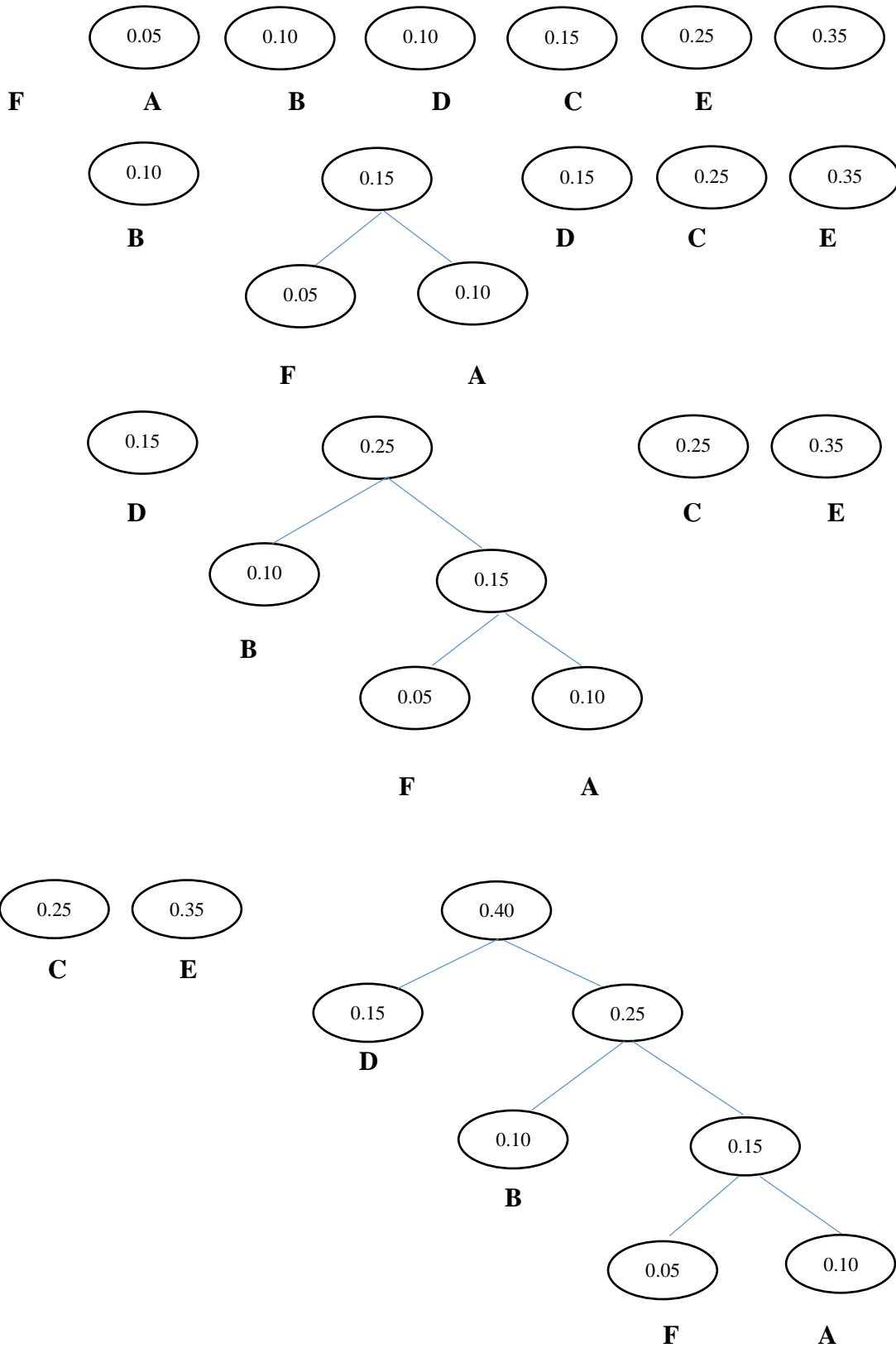
1- On trie les caractères par ordre croissant de probabilité

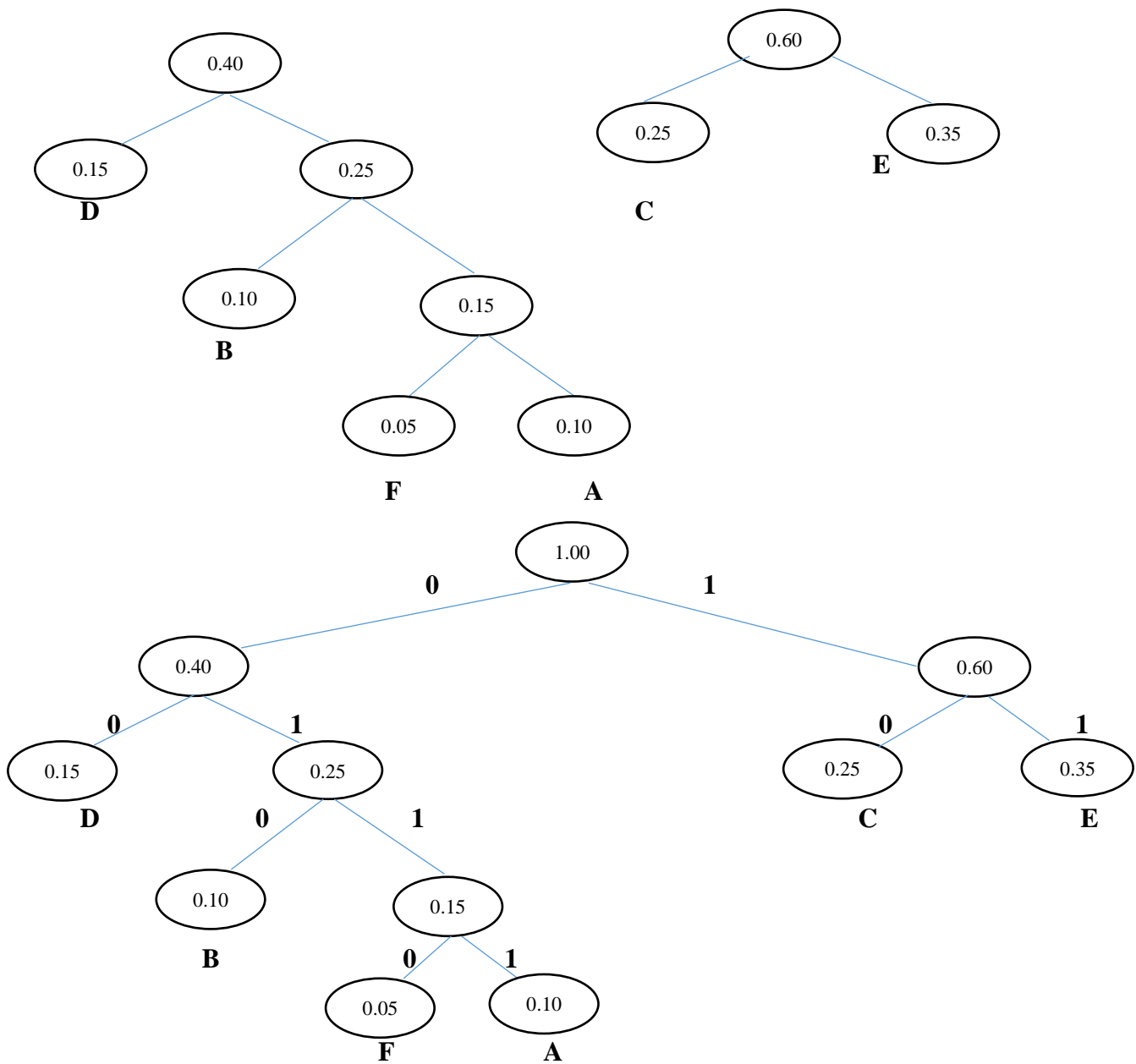
Symboles	F	A	B	D	C	E
Probabilité	0.05	0.10	0.10	0.15	0.25	0.35

Tableau 5 Symboles avec leurs probabilités par ordre croissant

2. On construit un arbre pour donner le code binaire de chaque caractère.

Construction de l'arbre : on relie deux à deux les caractères de probabilités les plus basses et on affecte à ce nœud la somme des probabilités des caractères. Puis on répète ceci jusqu'à ce que l'arbre relie toutes les lettres.





Le Codage des symboles est alors donné par le tableau3 :

Symboles	A	B	C	D	E	F
Code	0111	010	10	00	11	0110

Tableau 6 Mots de code des symboles

Avantages et inconvénients :**Avantages :**

✚ Code optimal à longueur variable produisant la longueur moyenne L des mots de code la plus faible

✚ Huffman permet d'avoir un taux de compression très élevé (50% en moyenne) et un temps de compression assez rapide.

Inconvénients :

✚ Le modèle est difficilement adaptatif. Ceci nécessite un recalcul complet de l'alphabet de sortie ce qui est coûteux en temps de calcul.

✚ Il nécessite la connaissance préalable des probabilités d'apparition des symboles de source.

C. Méthodes à base de dictionnaires (Lempel-Ziv-Welch) :

LZW (Lempel-Ziv-Welch), est un algorithme de compression adaptatif qui n'assume aucune connaissance des probabilités de symboles. [9]

La compression LZW fonctionne en lisant une séquence de symboles, en regroupant les symboles en cordes et en convertissant les chaînes en codes. Parce que les codes occupent moins d'espace que les chaînes qu'ils remplacent, nous obtenons une compression. Les fonctionnalités caractéristiques de LZW incluent.

II.7.2 Méthodes de compression avec perte d'informations

Les méthodes les plus importantes de compression avec perte sont :

- La réduction de l'espace des couleurs aux couleurs les plus fréquentes dans une image. Les couleurs choisies sont indiquées dans la palette de couleur de l'en-tête de l'image compressée. Chaque pixel indique juste une référence sur une couleur dans la palette de couleur.

- Le codage par transformation. C'est généralement la méthode la plus utilisée la transformé en cosinus discrète et la transformation par ondelettes sont les transformations les plus populaires. Le codage par transformation comprend l'application de la transformation à l'image, suivie d'une quantification et d'un codage entropique.

II.8 Mesure de qualité de la compression d'images

Dans un système de compression d'images, l'image compressée est toujours comparée à l'originale pour déterminer son rapport de ressemblance. L'évaluation peut être effectuée par plusieurs paramètres, parmi eux on peut citer :

✚ Taux de compression et débit binaire

C'est le but fondamental de la compression, il donne une mesure de performance des méthodes de compression des images fixes, autrement dit ; il donne une mesure de réduction

de la quantité d'information à transmettre. Le taux de compression se définit par :

$$T_{\text{taux}} = \frac{\text{nombre de bits dans l'image originale}}{\text{nombre de bits dans l'image comprimée}} \quad (\text{II.1})$$

✚ Qualité de reconstruction des images :

Pour les méthodes de compression sans perte d'informations, les images reconstruites sont exactement les images originales. Le problème de la qualité se pose pour les méthodes de compression avec perte d'informations. En effet, l'image reconstruite n'est pas exactement l'image originale, ainsi il nous faut une mesure de la qualité de l'image reconstruite.

Parmi ces fonctions, on trouve : [12]

1) EQM (Erreur quadratique moyenne) : Elle permet de mesurer la dégradation entre une image X origine et l'image X reconstruite après compression de coordonnées (n , m) et de dimension (N,M).

$$EQM = \frac{1}{M \times N} \times \sum_{n=1}^N \sum_{m=1}^M \left(X_{\text{origine}}(n, m) - X_{\text{reconstruite}}(n, m) \right)^2 \quad (\text{II.2})$$

Avec :

$X_{\text{origine}}(n, m)$ et $X_{\text{reconstruite}}(n, m)$: représentent les valeurs des pixels d'image d'origine et reconstruite.

2) SNR (Signal to Noise Ratio) : Il permet de mesurer le signal original sur bruit d'une image.

$$(\text{II.3}) = 10 \times \log_{10} \left(\frac{\frac{1}{M \times N} \times \sum_{n=1}^N \sum_{m=1}^M \left(X_{\text{origine}}(n, m) \right)^2}{EQM} \right) \text{dB}$$

3) PSNR (Peak Signal to Noise Ratio) : C'est le rapport signal crête sur L'erreur d'une image.

$$\text{PSNR} = 10 \times \log_{10} \left(\frac{(X_{\max})^2}{\text{EQM}} \right) \text{ dB} \quad (\text{II.4})$$

Avec :

X_{\max} : désigne la valeur du pixel maximale possible dans l'image qui prend une valeur 255 pour une image à niveau de gris codé sur 8 bits. Une valeur de PSNR infini correspond à une image non dégradée, et cette valeur décroît en fonction de la dégradation.

Entropie

Dans une image, L'entropie est une grandeur qui caractérise la quantité d'information que contient cette dernière. Par exemple une image dont tous les pixels ont la même valeur contient très peu d'informations car elle est extrêmement redondante, son entropie est faible. En revanche une image dont tous les pixels ont une valeur aléatoire contient beaucoup d'information, son entropie est forte. Donc l'entropie $H(s)$ d'une source simple $[S]$ associée à une loi de probabilité P est définie par la formule suivante :[11]

$$H(s) = - \sum_{i=1}^N P(s_i) \log_2(P(s_i)) \text{ bits} \quad (\text{II.5})$$

II.9 Etude du codeur (compresseur) d'images numériques JPEG

II.9.1 Présentation du codec JPEG

Le JPEG (Joint Photographic Experts Group) est un comité de standardisation pour la compression d'image dont le nom a été détourné pour désigner une norme en particulier, la norme JPEG, une norme voté comme internationale en 1992 ,elle satisfait à tous les besoins des différentes applications (médical, militaire, astronomique, ou journalistique) de la compression des images fixes (à niveau de gris et couleur), entre autres le stockage des images ainsi que l'échange (transmission) entre divers organismes, qui doit être plus facile et plus économique.

Parmi les choix proposés par la norme, on trouve des algorithmes de compression avec ou sans pertes (une compression avec pertes signifie que l'image décompressée n'est pas strictement identique à l'image d'origine) et différentes options d'affichage (séquentiel, l'image s'affiche en une passe pixel par pixel, ou progressif, l'image s'affiche en plusieurs

passes en incrustant progressivement les détails, ce qui permet d'avoir rapidement un aperçu de l'image, quitte à attendre pour avoir l'image entière).[12]

II.9.2 Fonctionnement général du JPEG

La compression d'une image au format JPEG 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 II.4) :

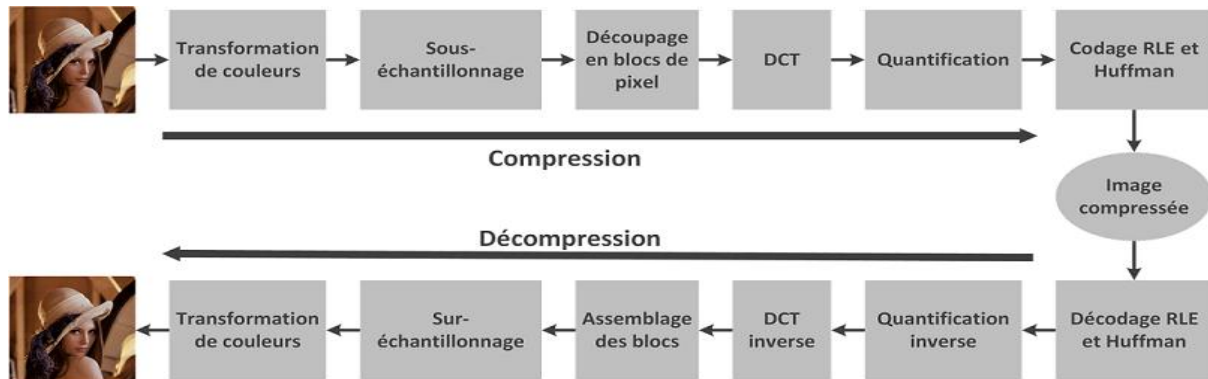


Figure II-4 Principe général de la compression/décompression JPEG

Nous allons maintenant décrire chacune de ces étapes permettant la compression. La décompression consistant à effectuer les mêmes étapes de façon inversée :

1. Transformation de couleurs

La première étape consiste à passer du modèle initial des couleurs de l'image (RVB) en modèle de type chrominance/luminance (YCrCb). Il est possible d'utiliser directement le modèle RVB, mais JPEG est plus efficace sur le modèle YCbCr où les composantes de chrominance (Cb et Cr) seront sous-échantillonnées afin d'aboutir à un modèle prédéfini.

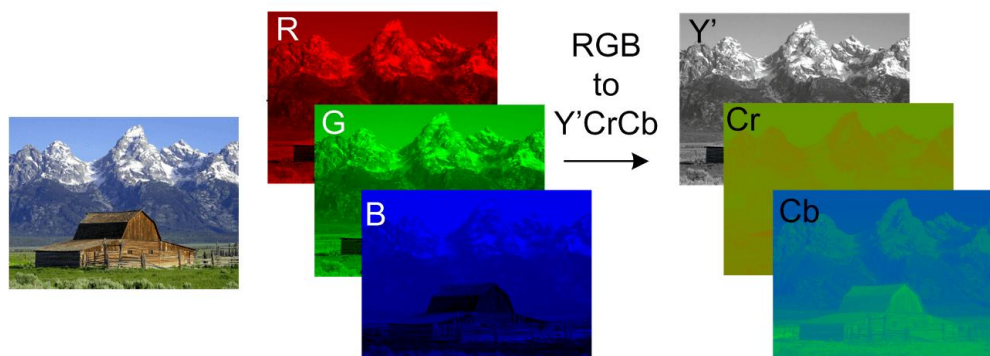


Figure II-5 Transformation de couleurs du RVB au Ycbcr

2. Le sous-échantillonnage

L'objectif ici va être de réduire l'information occupée par la chrominance comme le système visuel humain est moins sensible aux composantes chromatiques Cb et Cr qu'à la luminance, une perte d'information sur ce point passera relativement inaperçu.

En sous-échantillonnant horizontalement et verticalement les composantes chromatiques avant leur compression. La luminance est prise en chaque pixel tandis que la chrominance est prise comme une valeur moyenne pour un bloc de pixels.

Ainsi On distingue **3** types majeurs de sous-échantillonnages :

- **Le format 4 : 4 : 4** est le format de base où les composantes chromatiques n'ont pas été sous-échantillonnées. Aucune compression n'est effectuée et aucune perte de qualité ne peut être enregistrée.

Y	Y	Y	Y	Y	Y	Y	Y
Cb	Cb	Cb	Cb	Cb	Cb	Cb	Cb
Cr	Cr	Cr	Cr	Cr	Cr	Cr	Cr
Y	Y	Y	Y	Y	Y	Y	Y
Cb	Cb	Cb	Cb	Cb	Cb	Cb	Cb
Cr	Cr	Cr	Cr	Cr	Cr	Cr	Cr
Y	Y	Y	Y	Y	Y	Y	Y
Cb	Cb	Cb	Cb	Cb	Cb	Cb	Cb
Cr	Cr	Cr	Cr	Cr	Cr	Cr	Cr
Y	Y	Y	Y	Y	Y	Y	Y
Cb	Cb	Cb	Cb	Cb	Cb	Cb	Cb
Cr	Cr	Cr	Cr	Cr	Cr	Cr	Cr
Y	Y	Y	Y	Y	Y	Y	Y
Cb	Cb	Cb	Cb	Cb	Cb	Cb	Cb
Cr	Cr	Cr	Cr	Cr	Cr	Cr	Cr
Y	Y	Y	Y	Y	Y	Y	Y
Cb	Cb	Cb	Cb	Cb	Cb	Cb	Cb
Cr	Cr	Cr	Cr	Cr	Cr	Cr	Cr

- **Le format 4 : 2 : 2** consiste à supprimer les informations de la chrominance sur une colonne sur deux. La colonne restante possède alors la moyenne des chrominances des deux colonnes. Lors de la décompression, il suffira de recopier cette information sur la colonne vide.

Y	Y	Y	Y	Y	Y	Y	Y
Cb		Cb		Cb		Cb	
Cr		Cr		Cr		Cr	
Y	Y	Y	Y	Y	Y	Y	Y
Cb		Cb		Cb		Cb	
Cr		Cr		Cr		Cr	
Y	Y	Y	Y	Y	Y	Y	Y
Cb		Cb		Cb		Cb	
Cr		Cr		Cr		Cr	
Y	Y	Y	Y	Y	Y	Y	Y
Cb		Cb		Cb		Cb	
Cr		Cr		Cr		Cr	
Y	Y	Y	Y	Y	Y	Y	Y
Cb		Cb		Cb		Cb	
Cr		Cr		Cr		Cr	

• **Le format 4 : 2 : 0** effectue le même processus que le 4:2:2 mais en effectuant aussi cette opération sur les lignes. On divise donc ici par 4 l'information de la chrominance.

Y		Y		Y		Y	
cb	Y	cb	Y	cb	Y	cb	Y
Cr		Cr		Cr		Cr	
Y	Y	Y	Y	Y	Y	Y	Y
Y		Y		Y		Y	
cb	Y	cb	Y	cb	Y	cb	Y
Cr		Cr		Cr		Cr	
Y	Y	Y	Y	Y	Y	Y	Y
Y		Y		Y		Y	
cb	Y	cb	Y	cb	Y	cb	Y
Cr		Cr		Cr		Cr	
Y	Y	Y	Y	Y	Y	Y	Y
Y		Y		Y		Y	
cb	Y	cb	Y	cb	Y	cb	Y
Cr		Cr		Cr		Cr	
Y	Y	Y	Y	Y	Y	Y	Y

3. Le découpage en bloc de pixels

Le format JPEG, commence par découper chaque composante YCbCr en blocs carrés, Ces blocs sont des matrices de pixels 8*8 soient 64 pixels. L'algorithme de compression est calculé pour chaque bloc séparé, ce qui explique pourquoi ces blocs ou groupes de blocs deviennent visibles lorsque la compression est trop appliquée.

4. Transformation DCT

Cette étape consiste à appliquer une transformée en cosinus discrète (DCT) à notre matrice de pixels, Cette dernière permet de décrire chaque bloc en une carte de fréquences et en amplitudes plutôt qu'en pixels et couleurs, C'est l'étape qui coûte le plus de temps et de ressources dans la compression et la décompression , mais la plus importante car elle nous a permis de séparer les basses fréquences (des zones unies où les couleurs sont proches les unes des autres) et les hautes fréquences (des zones de contraste, de changement rapide dans les couleurs) présentes dans l'image. L'étape DCT lui-même est sans perte. La matrice résultante sera utilisée dans la prochaine étape : La quantification.

La transformée DCT s'exprime mathématiquement par : [12]

$$F(u,v) = \left(\frac{1}{4}\right) \cdot C(u) \cdot C(v) \sum_{i=0}^7 \sum_{j=0}^7 f(i,j) \cdot \cos\left((2 \cdot i + 1) \cdot u \cdot \frac{\pi}{16}\right) \cdot \cos\left((2 \cdot j + 1) \cdot v \cdot \frac{\pi}{16}\right) \quad (\text{II.6})$$

Et la transformée DCT inverse s'exprime par :

$$f(i,j) = \left(\frac{1}{4}\right) \sum_{u=0}^7 \sum_{v=0}^7 C(u).C(v).F(u,v) . \cos\left((2.i + 1).u.\frac{\pi}{16}\right) . \cos\left((2.j + 1).v.\frac{\pi}{16}\right) \quad (II.7)$$

Avec :

$$\begin{cases} C(u) = C(v) = \frac{1}{\sqrt{2}} & \text{pour } u = v = 0. \\ C(u) = C(v) = 1 & \text{autres } u \neq 0, v \neq 0. \end{cases} \quad (II.8)$$

F(u,v) : représente la valeur de la DCT au point de coordonnées (u, v) dans le bloc résultat de 8×8 pixels.

f(i,j) : représente la valeur du pixel de coordonnées (i, j) dans le bloc de l'image originale de 8×8 pixels.

Pour illustrer la transformée DCT nous avons pris un exemple d'un bloc de (8 × 8) pixels :

100	155	131	116	151	135	131	211	DCT →	145	-84	34	-69	4	-66	-35	72
120	135	127	88	155	131	155	179		-45	-28	28	19	10	-54	5	15
120	135	151	100	179	116	155	167		0	-2	-8	-15	-9	0	30	-41
120	155	151	108	191	112	155	179		9	-14	15	-11	5	8	-12	-32
135	151	135	210	197	112	179	179		1	1	3	-11	7	-23	-4	0
120	151	155	151	151	116	179	179		18	4	-17	-10	4	-10	7	-10
135	151	167	167	151	151	167	171		-5	1	-7	-20	1	-1	-3	5
120	151	179	151	151	131	155	167		3	1	1	9	2	7	2	-2

matrice 1
(Bloc de 8x8 pixels)
(domaine spatial)

matrice 2
(Tableau de 8x8 fréquences)
(domaine fréquentiel)

Fréquences
les plus
basses

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

matrice 2

Fréquences
les plus
hautes

5. La quantification :

La quantification est l'étape dans laquelle on perd réellement des informations (et donc de la qualité visuelle), mais c'est celle qui fait gagner beaucoup de place.

Elle a pour objectif d'atténuer les hautes fréquences (celles auxquelles l'œil humain est très peu sensible) d'une image qui ont été mis en évidence par la DCT. Et de garder les basses fréquences, c'est-à-dire celles auxquelles l'œil humain est très sensible.

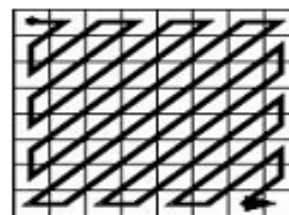
La technique est simple. Il suffit de diviser notre matrice (**matrice 2**) des fréquences avec la matrice de quantification (**matrice 3**) pour obtenir notre matrice quantifiée (**matrice 4**). Cette matrice a un avantage qui sera utilisée dans la prochaine étape : les valeurs non nulles ne sont toutes situées dans le bord haut-gauche de la matrice et Le résultat de la quantification des coefficients DCT sont parcouru en zigzag avant d'appliquer le codeur entropique qui est ici, Le codeur de Huffman qui compresse sans perte.

$$\begin{array}{c}
 \begin{bmatrix} 145 & -84 & 34 & -69 & 4 & -66 & -35 & 72 \\ -45 & -28 & 28 & 19 & 10 & -54 & 5 & 15 \\ 0 & -2 & -8 & -15 & -9 & 0 & 30 & -41 \\ 9 & -14 & 15 & -11 & 5 & 8 & -12 & -32 \\ 1 & 1 & 3 & -11 & 7 & -23 & -4 & 0 \\ 18 & 4 & -17 & -10 & 4 & -10 & 7 & -10 \\ -5 & 1 & -7 & -20 & 1 & -1 & -3 & 5 \\ 3 & 1 & 1 & 9 & 2 & 7 & 2 & -2 \end{bmatrix} \\
 \text{matrice 2}
 \end{array}
 \quad / \quad
 \begin{array}{c}
 \begin{bmatrix} 6 & 11 & 16 & 21 & 26 & 31 & 36 & 41 \\ 11 & 16 & 21 & 26 & 31 & 36 & 41 & 46 \\ 16 & 21 & 26 & 31 & 36 & 41 & 46 & 51 \\ 21 & 26 & 31 & 36 & 41 & 46 & 51 & 56 \\ 26 & 31 & 36 & 41 & 46 & 51 & 56 & 61 \\ 31 & 36 & 41 & 46 & 51 & 56 & 61 & 66 \\ 36 & 41 & 46 & 51 & 56 & 61 & 66 & 71 \\ 41 & 46 & 51 & 56 & 61 & 66 & 71 & 76 \end{bmatrix} \\
 \text{matrice 3}
 \end{array}
 \quad = \quad
 \begin{array}{c}
 \begin{bmatrix} 24 & -7 & 2 & -3 & 0 & -2 & 0 & 1 \\ -4 & -1 & 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\
 \text{matrice 4}
 \end{array}$$

6. Le codage

La dernière étape de la compression est le codage. L'objectif va être d'encoder notre bloc de pixel à l'aide du codage RLE suivit du codage de Huffman. Une fois l'opération appliquée à tous les blocs de pixels, nous obtenons un fichier compressé JPEG.

$$\begin{array}{c}
 \begin{bmatrix} 24 & -7 & 2 & -3 & 0 & -2 & 0 & 1 \\ -4 & -1 & 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\
 \text{matrice 4}
 \end{array}$$



Séquence zigzag

II.9.3 Algorithme de décompression JPEG

Procédé inverse de celui de la compression. Pour chaque bloc 8x8 de pixels :

- Décompression par la méthode de Huffman.
- Multiplication de la matrice obtenue par la matrice de quantification.
- Application de la DCT inverse pour retrouver une image plus ou moins dégradée

par rapport à l'image initiale.

- Suréchantillonnage
- La transformation de couleur (YCbCr en RVB).

II.9.4 Les avantages et inconvénients de la norme JPEG

a) Avantages :

Les principaux avantages de la compression JPEG sont : [4]

- Utilisation de beaucoup de paramètres, ce qui permet aux utilisateurs avancés d'expérimenter la compression/qualité désirée.

- Obtenir de bons résultats avec n'importe quelle image en tons continus, quel que soit la dimension, la couleur, la résolution ou autres caractéristiques de l'image.

- Une méthode de compression sophistiquée mais pas trop complexe, ce qui autorise des implémentations.

b) Inconvénients :

- JPEG offre très peu de résistance aux erreurs à cause du codage à longueur variable de type Huffman.

- Moins efficace sur les images géométriques

- Mauvaise qualité si la compression est trop poussée (pixellisation)

- JPEG est un format «avec perte» et est donc sujette à de nombreux problèmes de qualité visuelle des images.

Conclusion

Les images sont des documents très importants aujourd'hui, pour les utiliser dans certaines applications, ils doivent être compressés, plus au moins selon le but de l'application. Nous ne nous sommes intéressés au cours de ce chapitre qu'aux méthodes de compression des images qui effectuent cette compression de différentes manières comme la méthode de compression JPEG qui est le format le plus utilisées.

Le format JPEG est un standard largement répandue de compression d'image avec perte basé sur une transformée en cosinus discrète. La DCT qui fonctionne en séparant les images en parties de fréquences différentes (fréquences haute et fréquences basse), ces fréquences vont être quantifié au cours d'une étape appelée quantification, où une partie de la compression réellement se produit, les fréquences les moins importantes sont éliminées. Ensuite, seules les fréquences importantes restent et sont utilisés pour récupérer l'image dans le processus inverse (la décompression). En conséquence, les images reconstruites contiennent des distorsions de qualité. Le chapitre suivant fera l'objectif d'une discussion de quelques travaux parmi les plus récents et les plus répondus dans le domaine en question sur l'amélioration automatique de la qualité visuelle des images compressée

Chapitre III

Etat de l'art sur l'amélioration visuelle des images
compressées

Chapitre III Etat de l'art sur l'amélioration visuelle des images compressées.

III.1 Introduction

Comme vu durant les chapitres précédents, l'imagerie numérique est utilisée dans de nombreux domaines comme l'imagerie médicale, satellitaire ou encore industrielle. Dans la plupart de ces domaines, les images obtenues sont de mauvaise qualité ou de qualités insuffisantes. En effet ces images peuvent subir des dégradations comme du bruit ou après leur compression ou encore d'autres distorsions.

Nous pouvons traduire l'amélioration de la qualité d'une image numérique par le rehaussement d'image. Le rehaussement d'image intervient donc pour les rendre plus compréhensible et donc plus interprétable par l'œil humain. Ce dernier a pour objectif de compenser les différentes dégradations qu'a pu subir l'image, afin de retrouver l'image d'origine ou alors d'améliorer certains détails de l'image. Dans la partie qui suit, nous allons dresser l'état de l'art concernant les travaux scientifiques les plus récents pour donner aux lecteurs une vue générale sur l'actualité du domaine en question à savoir l'amélioration automatique de la qualité visuelle des images compressées.

Avant d'aborder l'état de l'art sur les méthodes d'amélioration de la qualité d'image rappelons d'abord quelques définitions de la notion de qualité et de fidélité.

III.2 Définir le terme de qualité

La perception désigne la sensation éprouvée par l'observateur à qui on présente un stimulus. C'est cette perception qui va lui permettre de porter un jugement positif ou négatif. Ce jugement peut porter sur l'ensemble de l'image ou plutôt s'axer sur une caractéristique précise. De plus, ce jugement peut être issu d'une comparaison entre deux données. Ces quelques remarques montrent la complexité du terme qualité et l'intérêt de définir correctement certaines notions. [2]

A. La notion de fidélité

Pour une image, la notion de fidélité permet de savoir si la reproduction d'une image est fidèle ou non à l'original. Dans ce cas, on met en place une mesure qui calcule la distance entre les deux images. Cette distance symbolise numériquement l'écart qu'il peut y avoir entre

les deux reproductions de l'image. Une image peut être fidèle mais de mauvaise qualité et une image de qualité correcte peut parfois avoir une fidélité assez mauvaise. [1]

B. La notion de qualité

La qualité d'image est un terme général employé par l'observateur pour évaluer la qualité inhérente de l'image. Elle est principalement subjective, signifiant que chaque observateur évalue la qualité d'image différemment. Certains peuvent penser qu'une image est d'assez haute qualité, mais on peut avoir un individu avec des niveaux plus élevés qui peut indiquer que l'image manque de qualité.

Avec des images numériques, beaucoup de personnes peuvent évaluer et classer des images par catégorie basées sur les bits de données ou les pixels. Par exemple, si on a deux images numériques, une d'un méga de pixels (1 million de pixels) et une image différente ayant 5 méga pixels d'information dans l'image, la plupart des personnes déclareraient que l'image avec plus de pixels a la qualité plus élevée. Et ce serait vrai dans un bon nombre de circonstances. Cependant, si l'image de 5 méga pixels était prise dans l'obscurité complète et l'image de 1 méga pixels était prise en utilisant la lumière régulière, la plupart des personnes conviendraient que bien qu'une image puisse avoir plus de méga pixels ne signifie pas nécessairement qu'elle a une meilleure qualité d'image. [1]

La qualité d'image pour la plupart est déterminée par la résolution et un équilibre de couleurs qui se nomme habituellement contraste. La résolution se rapporte à la quantité de détails dans une image. Habituellement plus il y'a de détails dans une image, plus la résolution est haute. [2]

Cette différence de notion entre fidélité et qualité peut respectivement être mise en parallèle avec la différence entre les notions de méthodes de reconstruction et d'amélioration d'images.

C. Evaluation de qualité

Après avoir introduit la notion de qualité, nous verrons qu'elle peut être évaluée soit de manière subjective par des observateurs donnant des jugements soit de manière objective qui cherche à déterminer la qualité des images algorithmiquement.

Evaluation subjectives :

Elle s'obtient à travers des tests subjectifs dont le processus est le suivant : un échantillon représentatif de la population juge la qualité d'une série d'images. Ces jugements sont

récupérés et analysés pour obtenir une note de qualité moyenne. La qualité subjective représente la vérité du terrain. [2]

Evaluation objectives :

Elle consiste à considérer la mesure de la qualité comme une évaluation quantitative de la perception des défauts et de la gêne qu'ils procurent sur le traitement des informations perceptuelles. Pour se différencier des mesures de fidélité, la mesure de qualité peut tenir compte des caractéristiques du système visuel humain. Pour être validée, une mesure objective de la qualité doit se rapprocher le plus possible des mesures de qualité subjectives obtenues pour un ensemble d'observateurs. [2]

III.3 La compression des images et ses artefacts

La réduction de la quantité d'information d'une image afin de pouvoir l'emmagasiner sans occuper beaucoup d'espace et la transmettre rapidement à travers les réseaux informatiques n'est atteint qu'en utilisant une compression avec pertes. Ces pertes peuvent se traduire par l'apparition de dégradations sur les images. Ces dégradations, plus communément appelées artefacts vont déterminer l'impact de la compression sur l'image perçue. Alors pour y remédier il faut savoir quantifier l'impact des artefacts de compression sur la qualité d'une image perçue et d'autre part essayer de minimiser cet impact en évitant, contrôlant ou supprimant ces artefacts. dans ce qui suit nous allons voir les conséquences des pertes causées par la compression sur les images.

a) L'effet de bloc

L'image est découpée en blocs. La quantification est effectuée sur chacun des blocs traités indépendamment les uns des autres. Si cette quantification est trop forte, lors de la reconstruction de deux blocs adjacents, il est probable que la continuité de l'information présente dans l'image non compressée soit cassée au niveau des frontières de blocs.

Une discontinuité de l'information apparaît alors au niveau de ces frontières de blocs et crée l'effet de bloc. La figure III.1 illustre cet effet, l'effet de bloc apparaît à une fréquence de 8 pixels correspondant au découpage de l'image par blocs. [3]

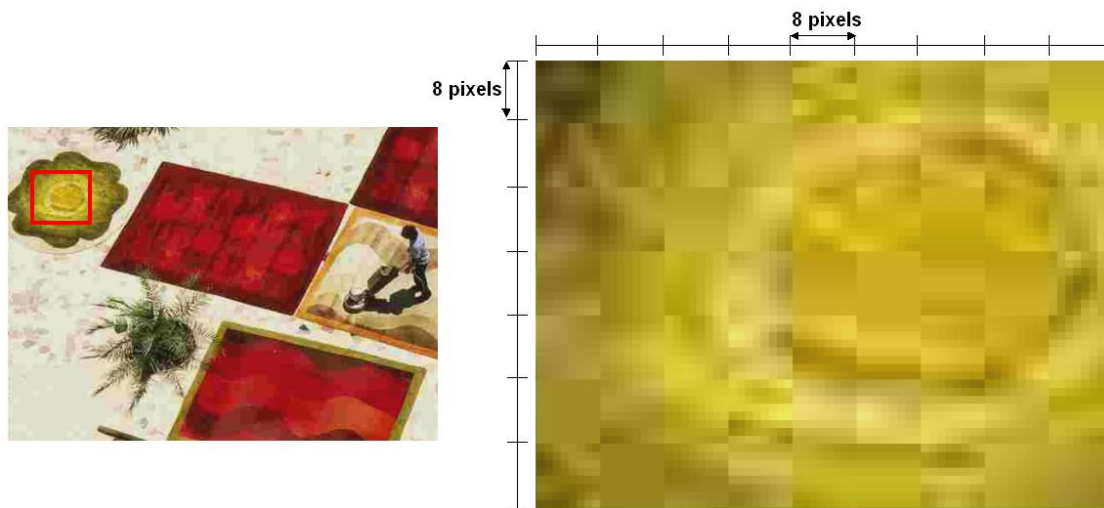


Figure III-1 L'effet de bloc.

b) Le flou

L'étape de quantification privilégie les coefficients basses fréquences aux coefficients hautes fréquences. Après quantification, toutes les informations liées au détail sont donc atténuées ou supprimées. [4]

Même si l'image reste compréhensible dans son contenu, les hautes fréquences représentant les contours ou les textures sont atténuées et provoquent un effet de flou sur l'ensemble de l'image (figure III.2).



Figure III-2 Image avec du flou

c) L'effet d'escalier :

Pour représenter un contour droit mais non vertical ou horizontal, l'idéal serait d'utiliser une combinaison des motifs hauts fréquences située à la fin (dans l'ordre du zig zag) de la matrice DCT. Or, ces coefficients sont généralement quantifiés et le contour est représenté par une combinaison de basses fréquences horizontales ou verticales. Lorsqu'un contour droit est représenté par des blocs adjacents, le contour sera reconstruit comme une succession de contours horizontaux et verticaux. L'effet d'escalier est représenté sur la figure III.3.

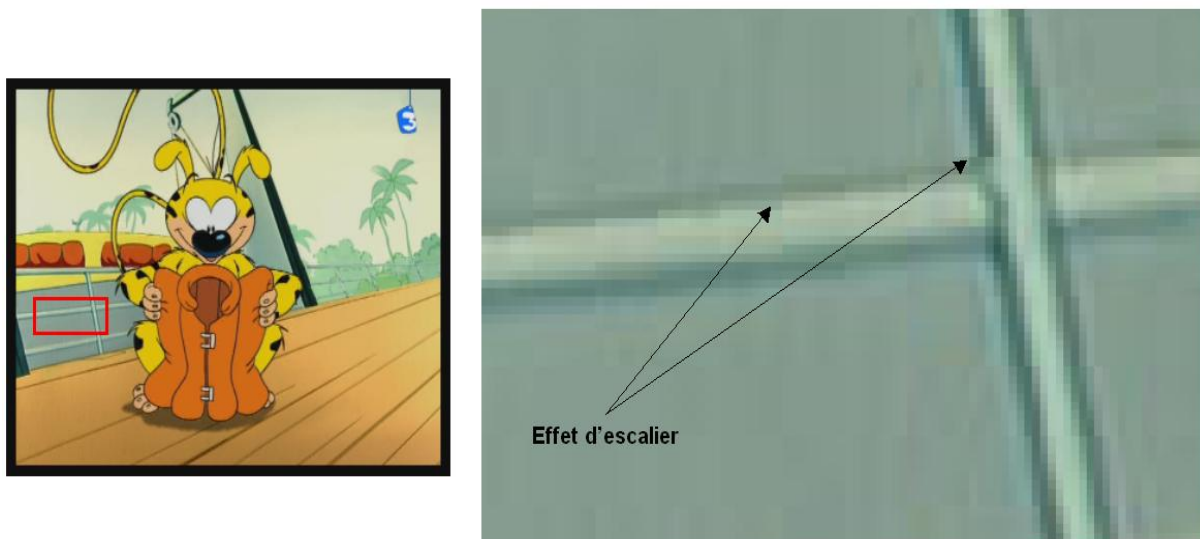


Figure III-3 L'effet d'escalier

III.4 Le rehaussement ou l'amélioration des images (Image Enhancement en Anglais)

L'amélioration des images est un domaine qui regroupe tout un ensemble d'opérations réalisées sur les images avec comme objectif général d'obtenir une image de meilleure qualité visuelle. Elle ne vise pas la fidélité de la scène mais la bonne qualité visuelle de l'image finale même si celle-ci est différente de l'originale.

III.4.1 Etat de l'art des méthodes d'amélioration d'images

Nous allons étudier les méthodes d'améliorations d'images proposées dans la littérature. Le but de ces corrections est de supprimer l'information non naturelle ajoutée par les artefacts. La suppression de cette information est donc principalement réalisée par différentes techniques.

Un large choix de méthodes a donc déjà été proposé pour éviter l'apparition de ces artefacts ou pour les corriger. C'est ce que nous proposons d'étudier dans la partie suivante :

A. Des méthodes inhabituelles

Que ce soit pour l'effet de bloc ou pour d'autres artefacts, la majorité des problèmes provient de la perte des coefficients hautes fréquences lors de la quantification. Dans [Niss, 1998], la technique consiste à retrouver ces coefficients hautes fréquences AC (coefficients alternatifs). En utilisant le coefficient DC (coefficients continu) d'un bloc donné et les coefficients DC des huit autres blocs voisins, Niss arrive à prédire par interpolation polynomiale les 5 derniers coefficients AC du bloc considéré. Cette méthode fait ses preuves dans les zones monotones mais donne de moins bons résultats lorsque les zones contiennent des contours car elles possèdent plus de coefficients AC. Néanmoins, si on tient compte de l'effet de masquage, c'est dans les zones avec peu de coefficients AC que les dégradations sont les plus visibles. La méthode nous semble donc très bien adaptée pour corriger les artefacts gênants pour le SVH (Système Visuel Humain). Par contre, le re-calcul des coefficients AC ne correspond pas à la contrainte de faible coût. [7]

B. Des techniques de filtrage linéaires et non linéaires

Le principe du filtrage consiste soit à supprimer de l'information gênante soit à faire ressortir

de l'information importante. Filtrer spatialement une image c'est modifier la valeur d'un pixel proportionnellement aux variations d'intensité lumineuse de ses voisins. Ces filtres sont utilisés pour augmenter certains contrastes, détecter des fronts ou des contours, ou au contraire réduire le bruit par lissage. Si $I(i,j)$ représente l'intensité du pixel I de coordonnées (i,j) dans l'image, les pixels entourant $I(i,j)$ peuvent être indexés comme indiqué sur la figure III.4 (dans le cas d'un voisinage 3×3). [8]

Un filtre linéaire affectera à $I(i,j)$ une valeur qui est une combinaison linéaire des pixels voisins suivant un voisinage donné. Un filtre non linéaire affectera à $I(i,j)$ une valeur qui n'est pas une combinaison linéaire des pixels voisins suivant un voisinage donné.

✚ Les filtres linéaires ou de convolution :

Une convolution est une fonction mathématique qui remplace chaque pixel par une somme

pondérée de ses voisins. La matrice définissant la taille du voisinage contient les poids respectifs de chaque pixel voisin. Cette matrice est aussi appelée masque ou noyau de

convolution. Filtrer un signal 2D (une image) consiste à convoluer sa fonction d'intensité I avec une fonction h appelée réponse impulsionnelle du filtre :

$$I'(i, j) = h(i, j) * I(i, j).$$

Pour supprimer les hautes fréquences représentées par les frontières de blocs, l'idée est d'utiliser un filtre passe-bas pour ne laisser passer que les basses fréquences. Par exemple, en utilisant le filtre moyenneur qui moyenne toutes les valeurs autour du pixel central, les hautes fréquences sont atténuées par le poids des basses fréquences.

✚ Les filtres non linéaires

En constatant que les frontières de blocs sont liées à l'information dans l'image, Ramamurthy

et al [Ramamurthy et Gersho, 1986] ont proposé en 1986 de remplacer les filtres linéaires par des filtres non linéaires pour pouvoir mieux tenir compte des contours et moins les lisser. En 1993, Hsu et al [Hsu et Chen, 1993] proposent un filtre médian non-linéaire à deux dimensions. En recherchant la valeur médiane des coefficients d'un noyau, ce filtre peut atténuer les pixels isolés d'une valeur très différente de leur entourage. Ce filtre médian non-linéaire a la caractéristique de bien préserver les contours mais reste toujours confronté au problème de différenciation entre les contours et les frontières de blocs. [8]

C. Amélioration par manipulation d'histogramme

Un histogramme permet d'avoir deux informations sur une image : [5]

- Savoir si l'image est sombre ou non. Une image sera sombre lorsque les pics sont sur plus vers la gauche de l'histogramme. S'ils sont vers la droite, l'image va être claire.
- Avoir des informations sur le contraste de l'image : si les pics sont au milieu de l'histogramme, l'image ne sera pas très contrastée. Si ceux-ci sont sur les bords, l'image sera fortement contrastée.

Les méthodes pour faire du rehaussement d'image par le traitement d'histogramme consiste en le calcul de l'histogramme de cette image. Il existe différentes méthodes :

Méthode 1 : Egalisation (ou linéarisation) d'histogramme

L'égalisation de l'histogramme consiste à équilibrer le mieux possible la distribution des

pixels dans la dynamique. L'idéal est d'obtenir un histogramme plat où l'on affecte le même nombre de pixels pour chaque niveau de gris (ceci étant impossible à réaliser pour les images numériques à cause de la nature discrète de l'histogramme et de la quantification des niveaux de gris (qui nous limite à un nombre fini de niveaux).

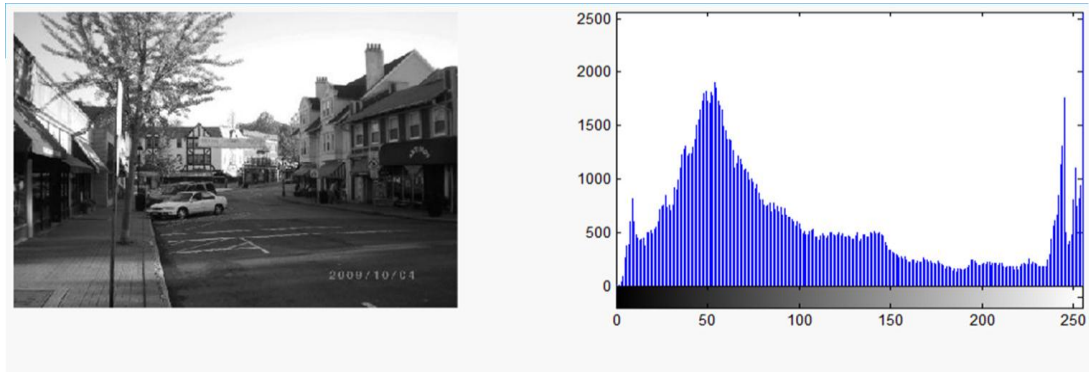


Figure III-4 Image avant égalisation d'histogramme

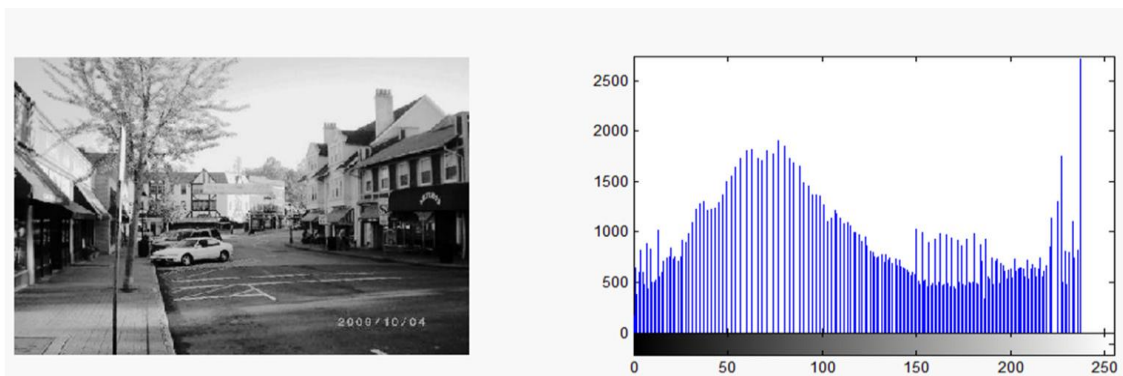


Figure III-5 Image après égalisation d'histogramme

Méthode 2: Normalisation (étirement) d'histogramme

Il s'agit d'une opération consistant à modifier l'histogramme, de telle sorte que l'image utilise toute la dynamique de la représentation. Ceci revient à étendre l'histogramme (répartition des intensités) afin que la valeur d'intensité la plus faible soit à zéro et que la plus haute soit à la valeur maximale. De cette façon, si les valeurs de l'histogramme sont très proches les unes des autres, l'étirement va permettre de fournir une meilleure répartition afin de rendre les pixels clairs encore plus clairs et les pixels foncés proches du noir. [8]

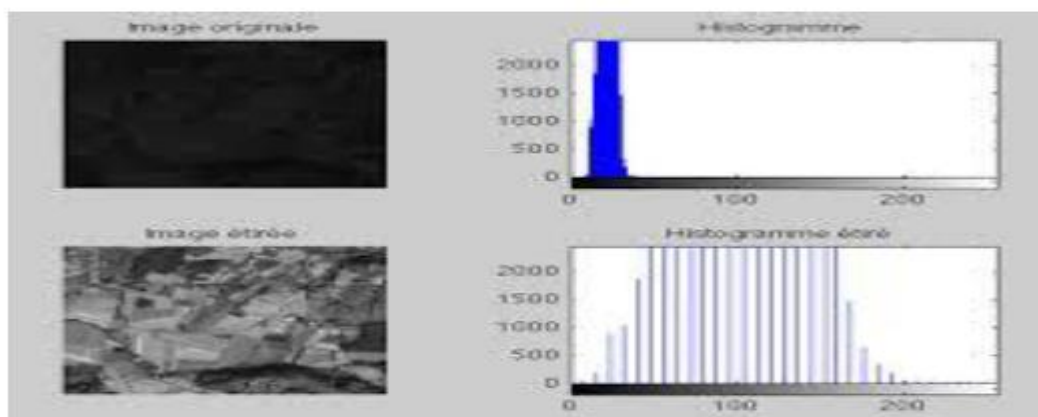


Figure III-6 L'étirement d'histogramme d'une image

D. Les méthodes de pré-traitements avant l'encodage

La compression des données favorise l'information à laquelle le SVH (Système visuel humain) est le plus sensible. Dans le même état d'esprit, [Ozkan et al., 1993] et [Kleihorst et al., 1995] proposent des méthodes de pré-filtrage. Ces méthodes suppriment le bruit en utilisant des techniques pour réduire l'information inutile. Cela permet de laisser plus de bits aux informations essentielles qui seront donc moins quantifiées. [6]

Dans le même registre, l'idée proposée dans [Kutka et al., 1996] est d'accentuer le contraste de l'image avant l'encodage. Ainsi, comme les dégradations génèrent souvent une perte de contraste, ces pertes seront compensées par l'étape de pré-traitement. Ce procédé est illustré sur la figure III.7

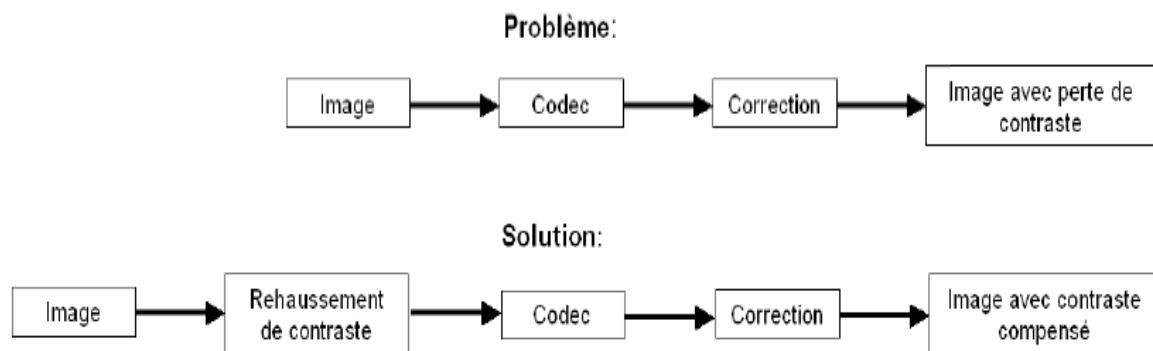


Figure III-7 Principe de la méthode de Kutka et al : pré-traitement par rehaussement du contraste

Dans un autre registre proche de la solution de pré-traitement proposée par Reeves et Lim [Reeve et Lim, 1984] sur le chevauchement des blocs, Nosraturia et al [Nosratinia, 2001] proposent de réappliquer le schéma de compression JPEG à l'image compressée. L'application étant réservée aux images JPEG, ils récupèrent les informations de compression se trouvant dans l'entête du fichier et appliquent cette même compression sur l'image décalée dans huit directions différentes.

Enfin, en moyennant toutes ces images, le résultat obtenu est l'image compressée dont l'effet de bloc est largement réduit. Cette idée, exploitée ici pour les images JPEG est réutilisée par Samadani et al [Samadani et al., 2004] qui proposent une amélioration de la correction de l'effet de ringing en modifiant la compression pour les pixels proches des contours. [5]

E. Les méthodes liées à l'encodage

✚ Les modifications des techniques d'encodage :

D'autres méthodes proposent une amélioration directement liée au système d'encodage pour éviter ou diminuer l'apparition des artefacts. Dans [Macq, 1992], la méthode proposée consiste à modifier les matrices de quantification en fonction du comportement du rapport signal sur bruit .

Dans [Macq et Shi, 1993] et [Nill, 1985], les auteurs proposent de combiner la transformée en cosinus discrète à un model basé sur le système visuel humain. Ces techniques, liées à notre perception, permettent de mieux quantifier les blocs selon leur contenu afin d'avoir le moins d'artefact possible.

Dans [Reeve et Lim, 1984], l'un des premiers articles proposés sur la réduction de l'effet de

bloc, Reeves et Lim décrivent une solution qui modifie le schéma classique de la compression par blocs. Comme l'effet de bloc est causé par la quantification indépendante des blocs, l'idée est de corriger le problème à la source en divisant l'image en plusieurs blocs se chevauchant. Certains pixels se trouvent alors dans plusieurs blocs et lors de la reconstruction de l'image, un pixel codé un ou plusieurs fois prend la valeur moyenne des valeurs codées. Cette méthode élimine les effets de blocs sans détériorer l'image mais augmente le nombre de bits en sortie nécessaire au codage de l'information car il y a beaucoup plus de blocs à traiter.

Ces techniques de modification de l'encodage impliqueraient de devoir modifier la majorité des encodeurs et décodeurs du marché. Même si la solution de résoudre le problème à la base semble la plus logique, elle n'est pas réalisable dans notre contexte.[7]

✚ Les corrections dans la boucle d'encodage :

L'intérêt d'introduire une correction dans la boucle d'encodage est d'améliorer la qualité des images de référence utilisées pour l'estimation de mouvement .A ce niveau, plusieurs types de méthodes ont été proposées. Dans [Pang et Tan, 1997], les auteurs proposent une faible correction au niveau des frontières donnant un meilleur PSNR mais insuffisant pour corriger efficacement les frontières de blocs.

Lee et Park dans [Lee et al. 1998] et [Lee et Park, 2001] proposent une méthode plus élaborée de filtrage. Leur méthode permet de corriger l'effet de bloc en se basant sur les coefficients DCT et le paramètre de quantification.

Dans le même état d'esprit, les auteurs [List et al. 2003] se basent sur les coefficients AC et DC des blocs, sur les paramètres du décodeur et proposent une correction efficace et robuste mais uniquement une correction de l'effet de bloc.

Ces méthodes rendent robustes les corrections. En effet, ces paramètres renseignent sur la force de compression et donc donnent une idée de la force de correction à appliquer ce qui explique l'intérêt des recherches portées sur ces méthodes jusqu'à aujourd'hui [Beermann et Wien, 2004],[Beermann et Wien, 2005], [Sun et al. 2006].

F. Les méthodes post-traitements après le décodage de l'image

Ces méthodes agissent directement sur les images décodées et n'ont accès qu'aux valeurs des pixels et parfois aux paramètres du décodeur. Ces traitements sont plus conséquents en temps de calcul qu'un traitement dans la boucle mais pourront toujours s'adapter sur les images quels que soient les paramètres de l'encodage. Cela augmente considérablement le domaine d'application de ce type de méthodes, et c'est pourquoi elles ont été majoritairement étudiées. On trouve d'ailleurs dans la littérature, des articles [Shen et Kuo, 1997] et [Nosratinia, 2001] entièrement consacrés à l'état de l'art de ces corrections.

Conclusion

Dans ce chapitre, nous avons présenté les travaux menés sur l'amélioration visuelle des images compressées, diverses techniques sont utilisées pour détecter et corriger les artefacts de compression. Dans le chapitre suivant nous allons présenter une description détaillée de notre application dédiée aux améliorations automatique de la qualité visuelle de images compressées basé sur le décodeur JPEG lui-même.

Chapitre IV

Implémentation et Tests

Chapitre IV : Implementation et Tests

IV.1 Introduction

Après les études théoriques et préliminaires menées précédemment nous allons entamer le volet pratique de notre projet et définir la mise en œuvre de notre implémentation, tout en commençant par une brève présentation des outils de développement utilisés suivie par une discussion des résultats obtenues concernant l'approche d'amélioration proposée de la qualité visuelle des images compressées JPEG

Partie I :

IV.2 L'environnement de programmation MATLAB

Le logiciel MATLAB, développé par la société The MathWorks, est devenu un langage de référence pour l'analyse et la résolution de problèmes scientifiques. Il intègre à la fois des solutions de calcul, de visualisation et un environnement de développement. Son nom provient de **MAT**rix **LAB**oratory). MATLAB a de nombreux avantages par rapport aux langages de programmation traditionnels (tel que le C/C++). Il permet le développement interactif par l'utilisation d'un langage interprété. La structure de données de base est le tableau. Il fournit de nombreuses fonctions préprogrammées regroupées en boîtes à outils (Toolbox) pour de nombreux domaines (signal processing, image processing, statistics, control theory, optimization, ...). De plus, MATLAB dispose d'une excellente documentation à travers le HELP.

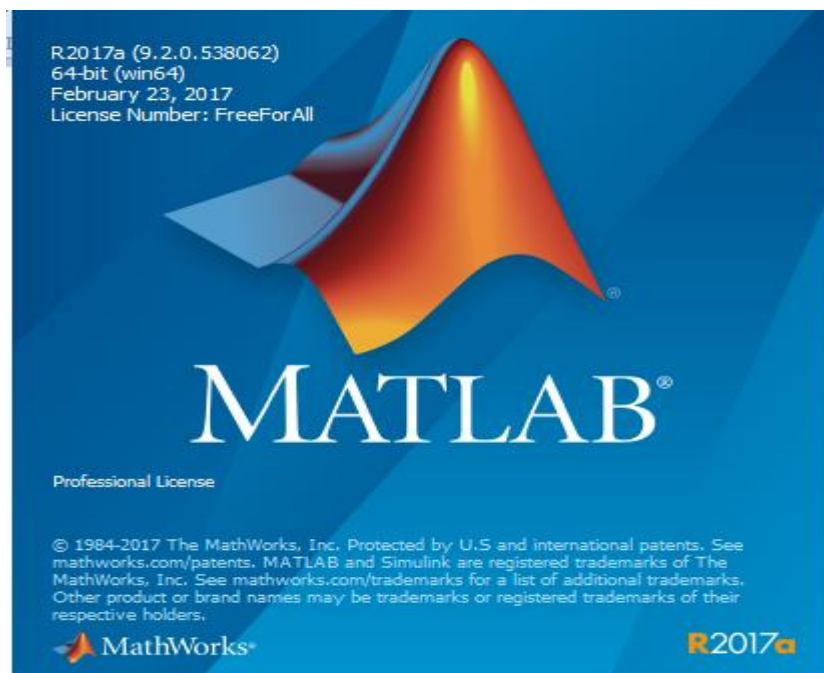


Figure IV.1 : l'environnement MATLAB.

1. Intérêts :

- Programmation infiniment rapide pour le calcul et pour l'affichage.
- Une librairie très riche.
- Possibilité d'inclure une programmation en C/C++.
- Langage interprété : Pas de compilation donc pas d'attente pour compiler.
- Code facile à comprendre et très lisible.
- Une aide très bien faite.

2. Inconvénients :

- Vitesse de calcul moins rapide qu'en C/C++.
- Application auto-exécutable peu pratique.

IV.3 Image Processing Toolbox :

Propose un ensemble complet d'algorithmes standard de référence et d'applications pour le traitement d'images, l'analyse, la visualisation et le développement d'algorithmes. Les opérations qui peuvent être effectuées sont la segmentation des images, l'amélioration des images, la réduction du bruit, les transformations géométriques, le recalage des images et le traitement des images en 3D.

Les applications d'Image Processing Toolbox permettent d'automatiser les processus courants de traitement d'images, de manière interactive, on peut segmenter des données d'images, comparer des techniques de recalage d'images et traiter par lots de grandes quantités de données. Les applications et les fonctions de visualisation permettent d'explorer des images, des volumes 3D et des vidéos, d'ajuster le contraste, de créer des histogrammes et de manipuler des régions d'intérêt.

IV.4 Analyse des besoins

l'objectif est de développer une application capable d'améliorer la qualité visuelle des images compressées par le standard JPEG tout en essayant de résoudre son problème majeure qui est la dégradation de la qualité d'image, on se basant sur deux approches différentes

✚ Notre approche :

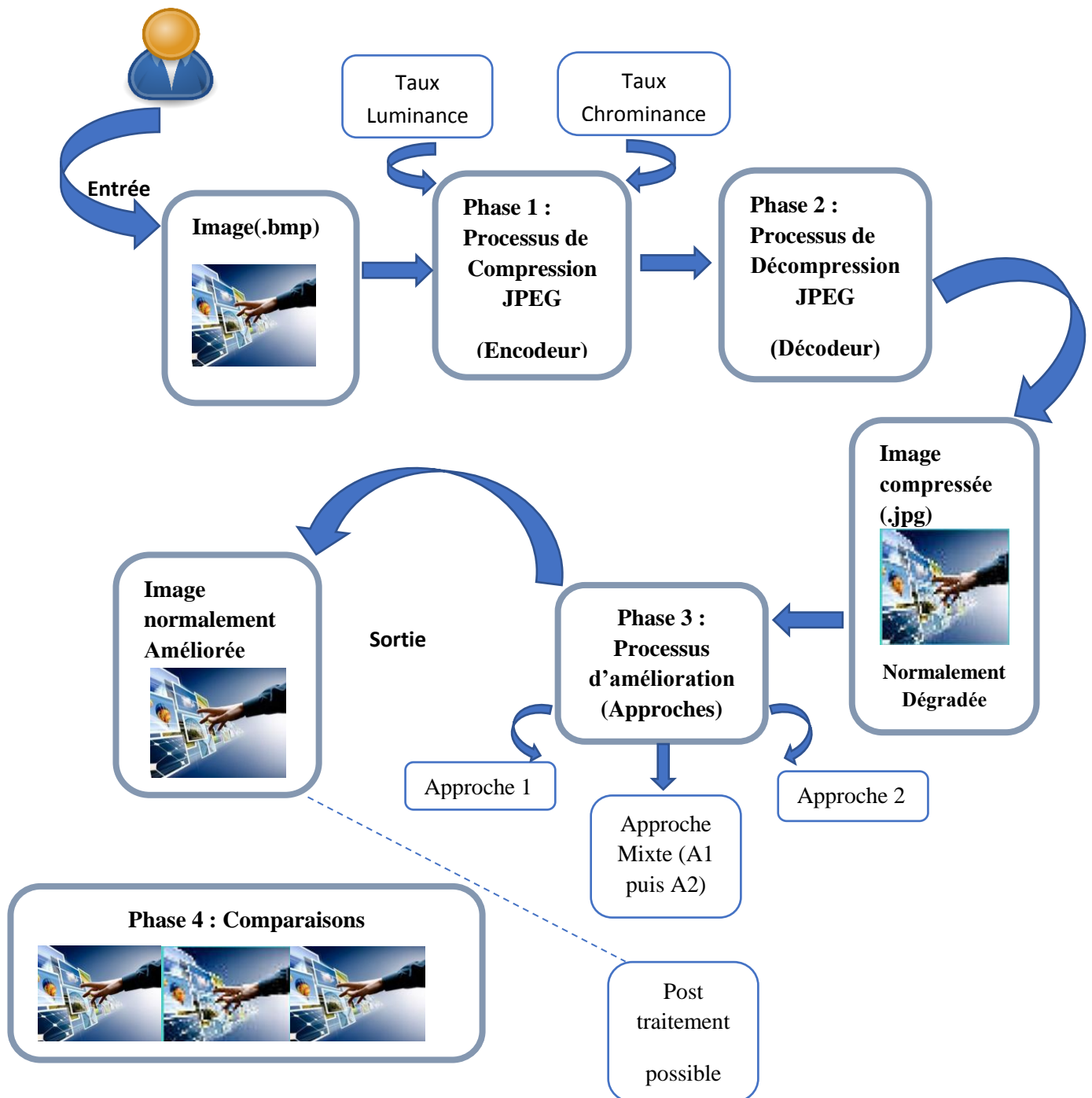
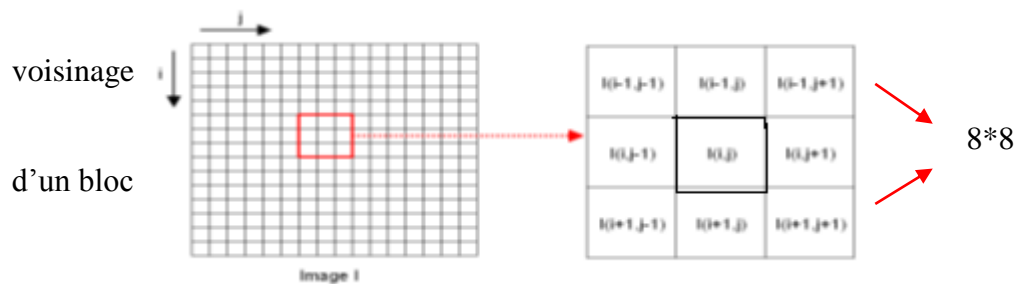


Figure IV-1 Présentation générale de l'approche proposée

Le but de l'amélioration est de rendre les images plus apte à l'interprétation humaine ou à celle de la machine, différentes méthodes ont été proposées, dans le cadre de notre travail on propose d'effectuer d'éventuelles modifications au niveau du domaine spatiale sous forme d'un petit programme MATLAB Qu'on explique brièvement dans la partie suivante :

Approche 1 :

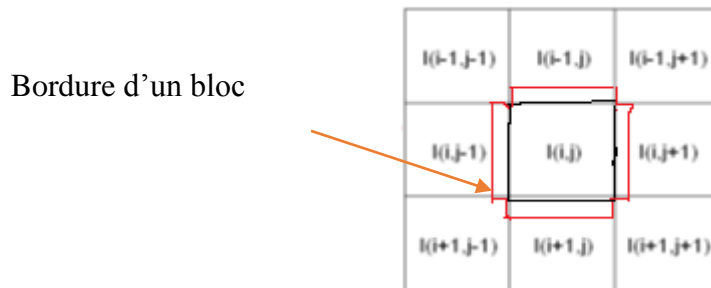
- Ajouter un bord 8*8 noir à l'extérieur de l'image d'entrée en utilisant la fonction zeros afin de créer un voisinage pour les blocs situés au bord de l'image originale.
- Découper l'image en blocs 8*8 selon la DCT
- Calculer l'écart type (le contraste) global de blocs 24*24 (connectivité 8)



- Calculer l'écart type de voisinage (le contraste de voisinage)
- Calculer la différence entre écart type global et l'écart type
- Chaque calcul précédent a un seuil (dans l'approche 1 on a trois seuils, seuil 1, seuil 2, seuil 3)
- Pour faire le traitement il faut que le contraste global soit supérieur au seuil 1 c'est-à-dire visualiser une dissemblance dans l'image et le contraste de voisinage est plus petit

Approche 2 :

- Faire la différence entre les bordures d'un bloc 8* 8 et son voisinage.



- Comparaison avec le seuil 4
- Appliquer un traitement de lissage si le critère est satisfaisant par rapport au seuil

IV.5 Conception & modélisation

Consiste à créer une représentation simplifiée d'un problème. En utilisant une méthode de modélisation UML (Unified Modeling Language) qui utilise une représentation graphique. L'usage d'une représentation graphique est un complément excellent à celui de représentations textuelles. En effet, l'une comme l'autre sont ambiguës mais leur utilisation simultanée permet de diminuer les ambiguïtés de chacune d'elle. Un dessin permet bien souvent d'exprimer clairement ce qu'un texte exprime difficilement et un bon commentaire permet d'enrichir une figure. Parmi les diagrammes on a utilisé le diagramme de classe.

A. Diagramme de classe

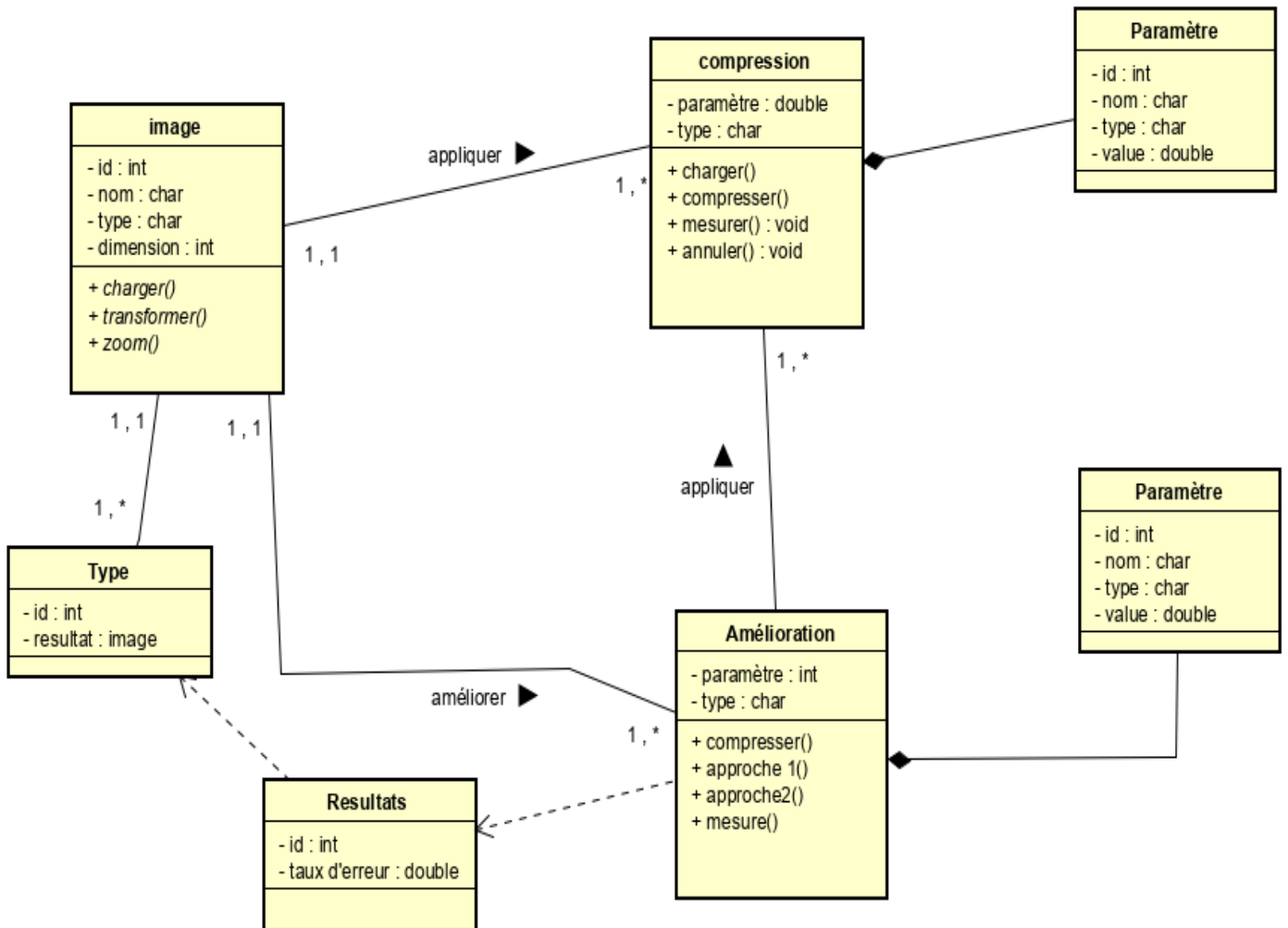


Figure IV-2 Diagramme de classe

Partie II : implémentation

IV.6 Présentation de l'application implémentée



Figure IV-3 Page d'accueil de l'application

Le Bouton « à propos » donne des informations sur le projet :



Figure IV-4 Informations sur le projet

Le bouton « Quitter » de notre application permet de fermer une interface :

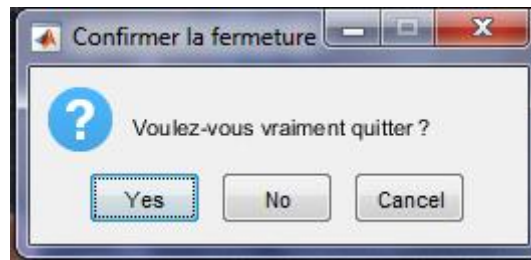


Figure IV-5 Fermeture d'une d'interface

Le bouton « Menu » de notre application se compose de cinq opérations essentielles :

- a. opérations de base.
- b. Prétraitement
- c. Compression JPEG
- d. Amélioration JPEG
- e. Autres méthodes d'amélioration

Chacune de ces options citées précédemment nous conduit vers une nouvelle interface pour appliquer un traitement spécifique.

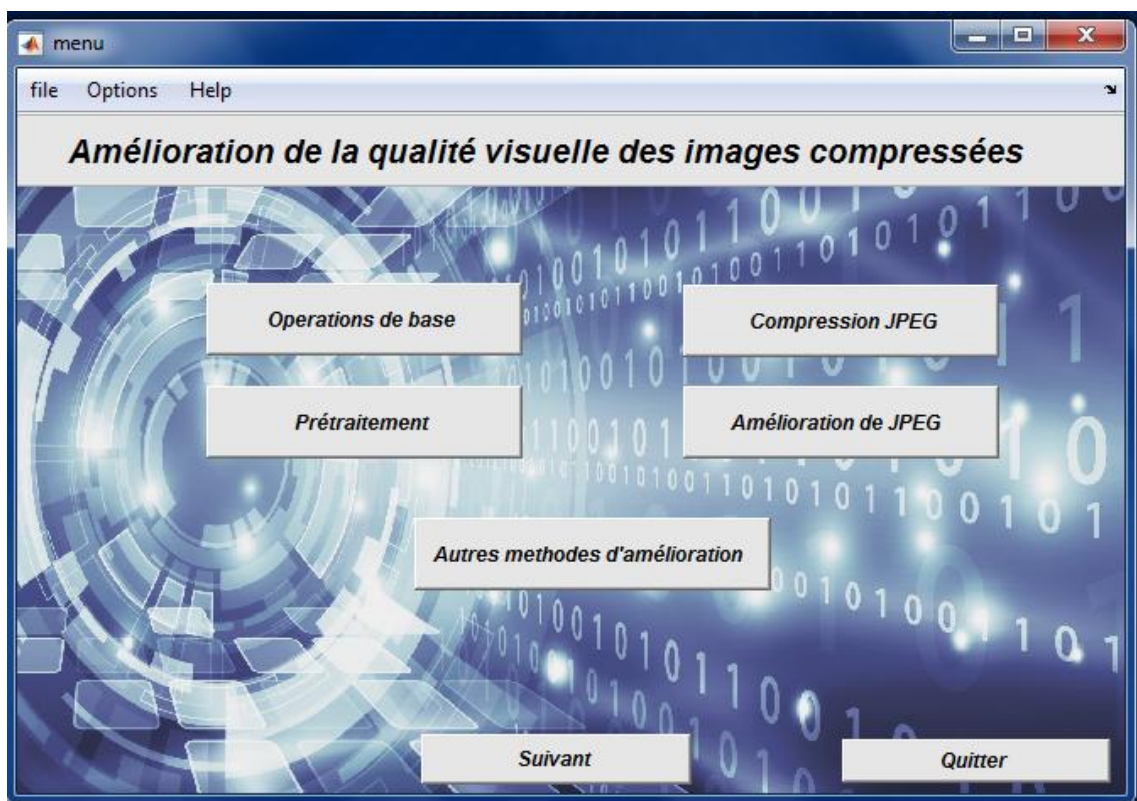


Figure IV-6L l'interface menu principale de l'application

a) opérations de base :

A partir de cette fenêtre nous pouvons réaliser certaines opérations de bases en général sur une image. Comme par exemple la séparation des composantes d'une image couleur (R, V, B), mesure de distance, informations sur le pixel etc.....



Figure IV-7 Fenêtre d'opération de base d'une image.

b) Prétraitement :

Préparer l'image au traitement ultérieurs pour mettre en relief l'information utile contenu dans l'image, permettre une meilleure visualisation de l'image (éliminer le bruit, filtrer l'image, détecter contours etc.....)

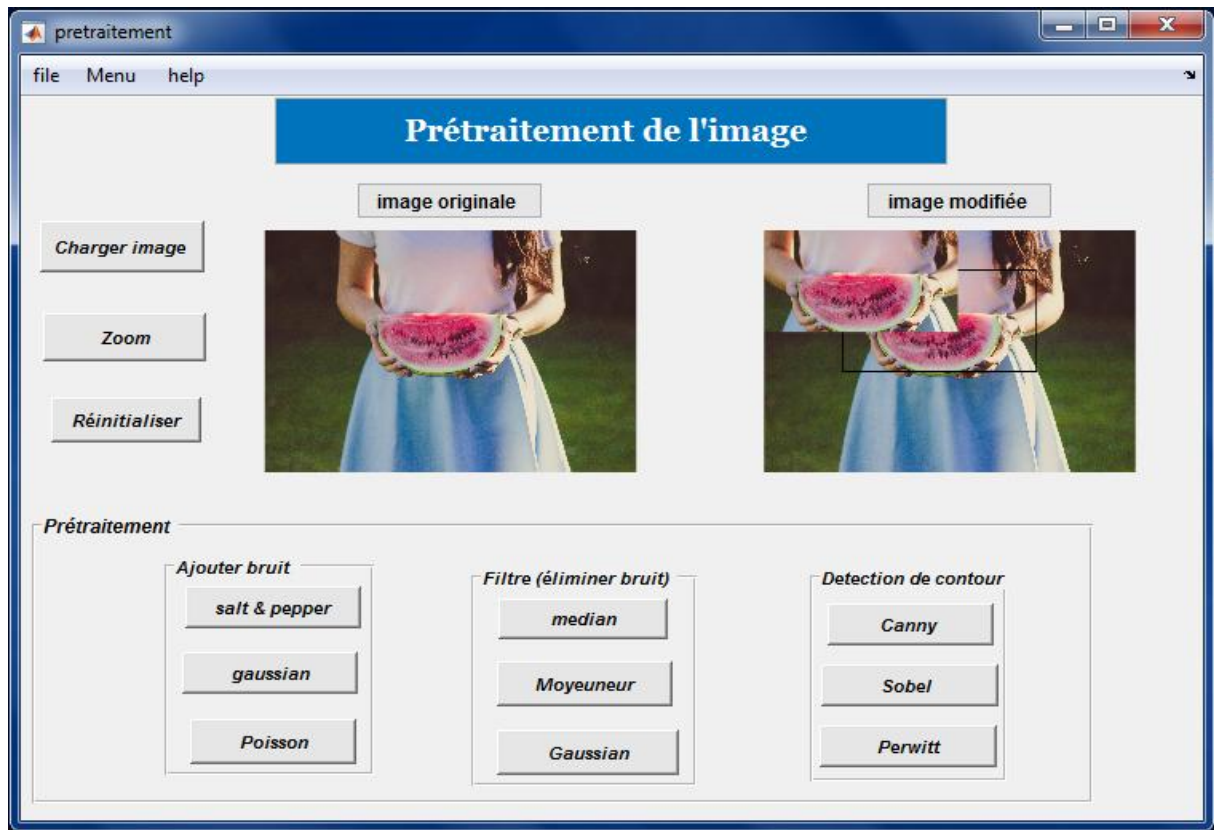


Figure IV-8 Fenêtre de Prétraitement de l'image

c) Compression :

A partir de cette fenêtre nous pouvons effectuer le procédé de compression JPEG sur des images en couleur et niveaux de gris et faire une évaluation de la qualité des images après compression, on mesure la qualité de reconstruction de l'image par rapport à l'image originale on utilisons une grandeur de distorsion qui est l'erreur quadratique moyenne MSE (Mean Square Error).

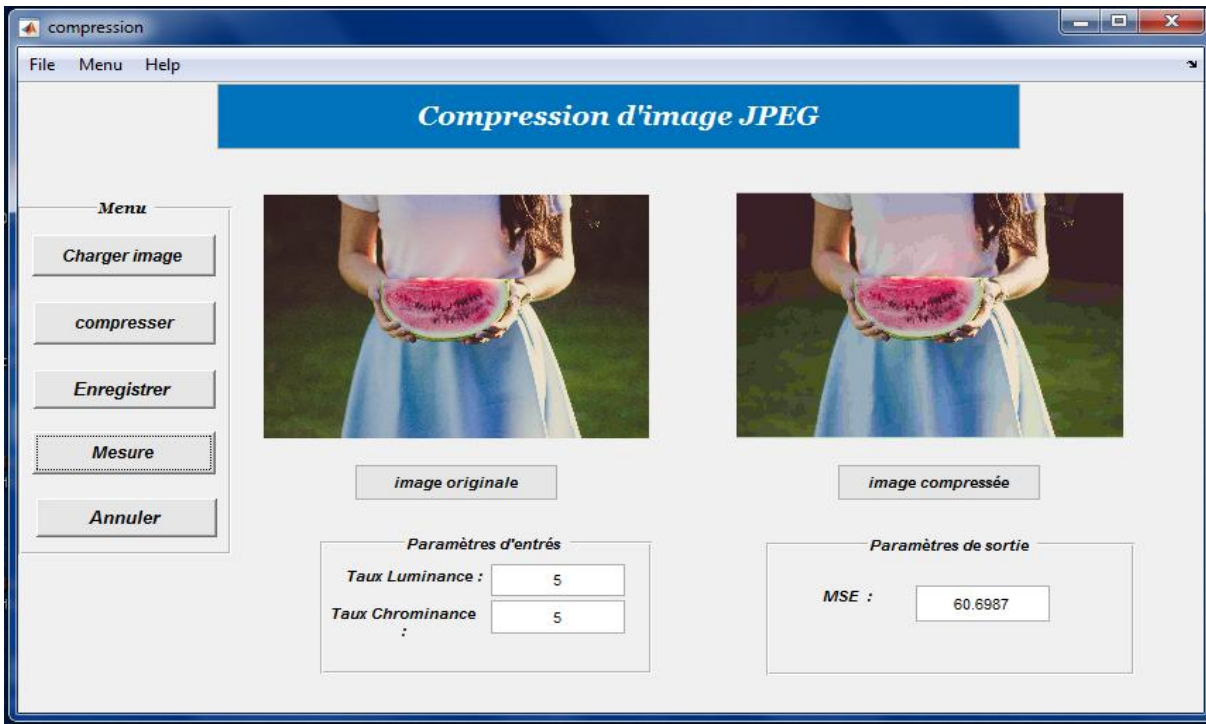


Figure IV-9 Fenêtre de compression d'image

d) Amélioration JPEG

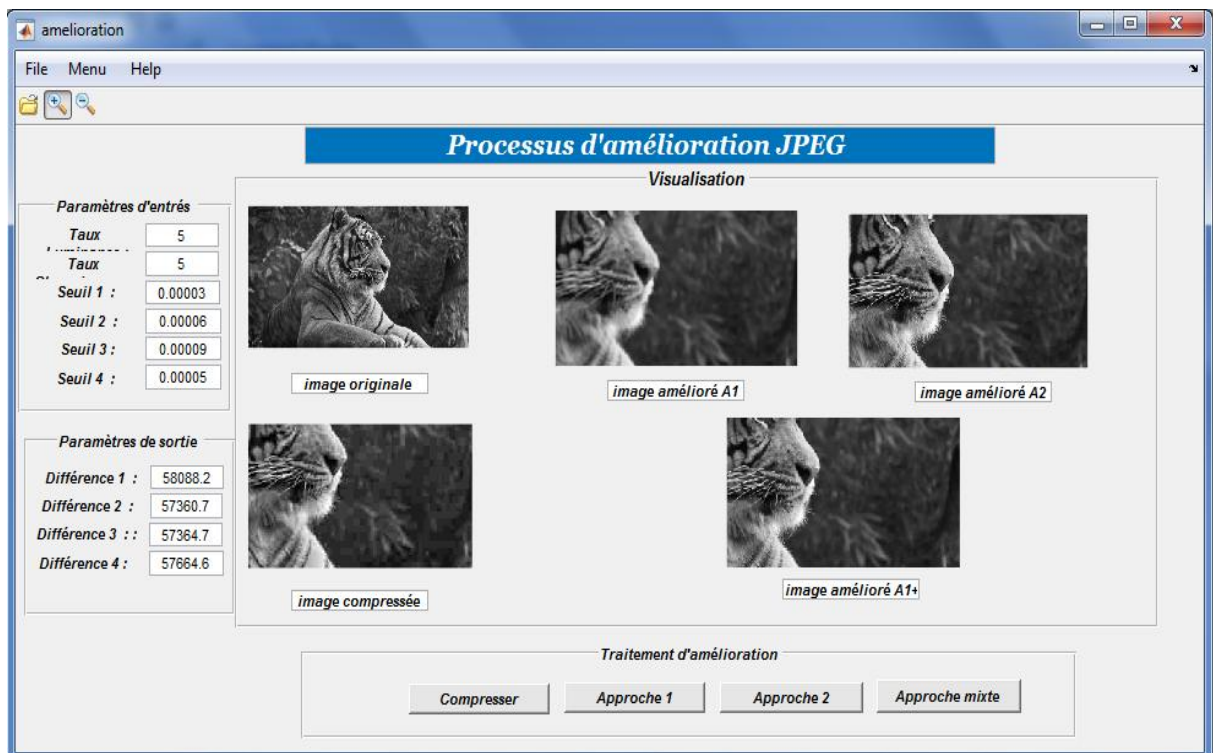


Figure IV-10 Fenêtre d'amélioration d'image avec l'approche

e) Autres méthodes d'amélioration :

l'amélioration d'image consiste à modifier les caractéristique visuelle de l'image (contraste) pour faciliter son interprétation par l'œil humain il peut s'agir par exemple de rehausser le contraste, d'égaliser un histogramme etc. ...



Figure IV-11 Fenêtre d'amélioration d'image avec d'autres méthodes

- Rehaussement de contaste :

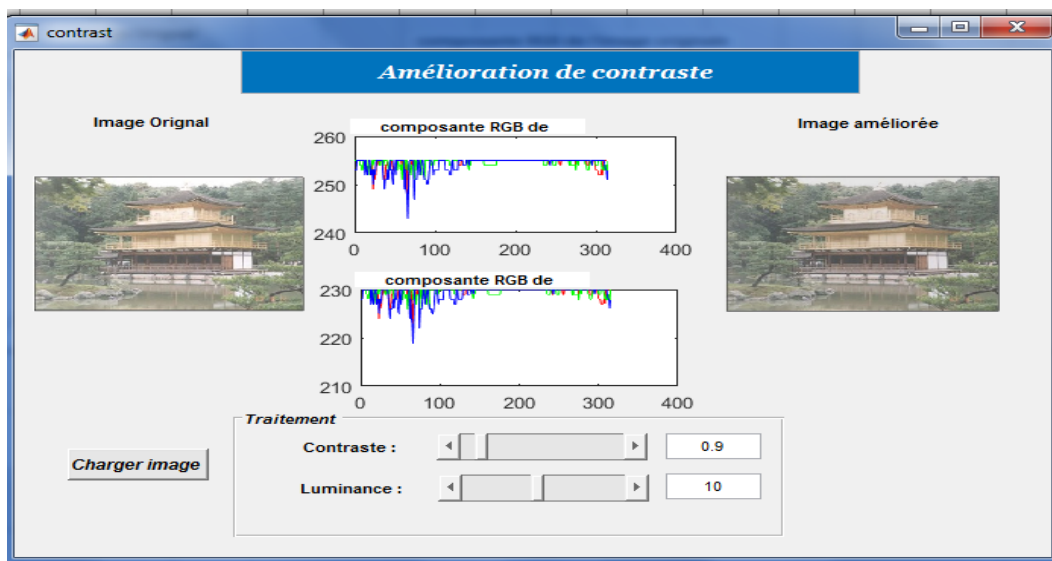


Figure IV-12 Fenêtre d'amélioration de contraste

IV.7 Résultats et discussion de l'approche proposée

A travers cette partie, nous allons procéder à une succession d'expériences selon des paramètres. On prend des images en niveaux de gris pour tester l'approche proposée en changeant à chaque fois les seuils et on affiche le résultat

a) Comparaison entre l'image compressée et l'image améliorée par l'approche 1



a) Image compressée TL = 5



b) Image améliorée approche 1

seuil1=0.0005, seuil 2=0.0015, seuil =0.00001

b) Comparaison entre l'image compressée et l'image améliorée L'approche 2



a) Image compressée TL =5



c) Image améliorée approche 2

seuil 4 = 0.00026

c) Comparaison entre approche 1 et approche 2 :



b) Image améliorée approche 1



c) Image améliorée approche 2

Figure IV-13 Résultat d'application des deux approches

d) Comparaison entre l'image compressée et l'image améliorée approche mixte

Image compressée TL = 5 Image améliorée approche mixte

❖ Discussion de résultats :

- Plus la valeur de seuil est basse, meilleure est la qualité
- la valeur des seuils doivent être plus petite lors de test afin d'obtenir une qualité mieux .

❖ Remarques importante :

- Les images utilisé pour tester les deux approches sont d'origine des images de formats .bmp en niveaux de gris

Conclusion

A travers ce chapitre nous avons pu présenter notre travail sous forme d'une application qui sera utilisée par les futurs étudiants pour mener plus d'expériences peut être validé notre point de vue concernant les difficultés rencontrées lors du choix des seuils des deux approches

Nous pouvons dire que l'approche proposée est moins efficace mais reste à améliorer en utilisant la logique floue pour trouver la meilleure configuration des seuils.

Conclusion Générale et Perspectives

Conclusion générale et perspectives

La compression des données et particulièrement des images date déjà de plusieurs décennies. Cependant, elle est toujours d'actualité compte des exigences technique et technologique liées aux multimédias d'une manière générale et aussi aux moyens de communications comme les réseaux informatiques. De nombreux algorithmes ont vus le jour et certains sont devenus des normes et des standards. Les techniques de compression avec pertes assurent généralement des taux de compression relativement élevés mais avec un certain degré de dégradation. Les objectifs des nouvelles techniques de correction des artefacts de compression avec perte est donc de réduire plus ces dégradations et d'altérer les moins possible. Parmi les techniques de compression avec perte nous avons les méthodes basées sur les transformations comme par exemple le rehaussement de contraste, l'égalisation d'histogramme, la correction gamma etc..

A travers ce mémoire nous avons essayé en premier lieu d'introduire les notions fondamentales de l'image numérique. En deuxième lieu nous avons abordé la notion de compression d'image avec ces différentes méthodes réversible et irréversible.

En troisième lieu une brève discussion a été l'objet d'une illustration des travaux parmi les plus récents du domaine en question.

Finalement le quatrième chapitre a fait l'objet d'une implémentation d'une approche basé sur des calculs de contraste en fixant des seuils pour faire une comparaison afin de mener une certaine amélioration de qualité d'image compressée.

Nous avons donc pris un exemple assez simple d'un algorithme classique à savoir le JPEG et nous avons essayé de réduire quelques artefacts dans le but de rendre l'image interprétable par l'œil humain.

Ce projet de fin d'étude représente réellement une plateforme très modeste pour le rehaussement de la qualité visuelle d'images compressées qui constitue une voie de recherche **d'actualité** tout en espérant que cette application sera enrichie par d'autres approches basé sur la logique floue puisque le choix de la bonne configuration des seuils demande beaucoup de travail et enfin extraire les règles générale.

Bibliographies

Chapitre I : l'image Numérique

- [1] : chapitre 1 cours de traitement d'image de Mr.Mezzoug Karim ,page2
- [2] : chapitre 3, analyse et traitement d'images numériques -Bres, S., Jolion, J. M., & Leb F. Traitement et analyse des images numériques (p. 19). Hermes Science Publications.
- [3] Zeroual Djazia, « implémentation d'un environnement parallèle pour la compression d'image à l'aide des fractales » mémoire de magister ,université biskra 2006
- [4] Cours traitement d'image ,DUT Informatique (Université de Strasbourg),pp9
- [5] chapitre 3 cours de traitement d'image de Mr.Mezzoug Karim ,page3
- [6] Benfriha Sara « Segmentation d'image par cooperation region contour »,mémoire master professionnel, université kasdi merbah ouargla 2015
- [7] chapitre 5 cours de traitement d'image de Mr.Mezzoug Karim ,page6
- [8] chapitre 2 cours de codage et transport de flux multimédia de M^r Ghafour yacine

Webographie

- [w1] : www.lmgDist/LM.fr/travs/C1, K.chakib Généralité sur le traitement d'image
- [w2] : <https://fr.wikipedia.org> ,Encyclopédie Universalis, « IMAGE NUMÉRIQUE ET IMAGE DE SYNTHÈSE »
- [w3] http://www.kolor.com/wiki-fr/Dynamique_d%27une_image,
- [w4] <https://patrick-bonnin.developpez.com>,

Chapitre II : La Compression et la Décompression des Images Numériques

[1] thèse présenté à la faculté des arts et science en vue de l'obtention du grade philosophie docteur en informatique, intitulé « Contribution à la compression des données » .

[2] Z.Djazia (2006).Implémentation D'un Environnement Parallèle Pour La Compression D'image A L'aide Des Fractales. Université de Batna.

[3] Chapitre III compression d'images fixes

[4] PIGEON, Steven. Contribution à la compression de données. Thèse de Doctorat : Informatique. Montréal, 2001.

[5] N. Morceau, “ Techniques de compression des signaux ”, Edition Masson, 1995.

[6] Catherine LAMBERT-NEBOUT . 'La compression embarquée d'images pour les systèmes optiques d'observation spatiale'.

[7] N. Morceau, “ Techniques de compression des signaux ”, Edition Masson, 1995.

[8] D. A. Huffman, “ A method for the construction of minimum - redundancy codes”

[9] Hari M, “Compression Algorithms: Huff man and Lempel-Ziv-Welch (LZW)”, 6.02 Introduction to EECS II: Digital Communication Systems Fall 2012.

[10] C. Benoit & A. Dusson, "La compression de données informatiques". Juin 1999 <http://www.esil.univ-mrs.fr/~cbenoit/projets/comp/>

[11] F. Davoine, Compression d'Images par Fractales Basée sur la Triangulation de Delaunay, Thèse l'INPG, Institut National Polytechnique de Grenoble,

[12] SAHIR MOURAD. Compression des images numériques par la technique des ondelettes Mémoire de magister : Sciences en Electronique. Setif, 2011

Webographie

[w1] http://igm.univ-mlv.fr/~dr/XPOSE2013/La_compression_de_donnees/wit.html, Divers tutoriaux du site "Comment ça marche".

[w2] <https://fr.wikipedia.org/wiki/Compression>

[w3] <https://guersanguillaume.com/compresser-image-en-ligne/>

[w4]<https://blog-fr.orson.io/web-design/pourquoi-compresser-les-images-sur-mon-site-internet>

Chapitre III : Etat de l'art sur l'amélioration visuelle des images compressées

- [1] C. Charrier, C. Larabi, H. Saadane. Evaluation de la qualité des images. Ecole d'hiver sur l'image numérique couleur, 2005.
- [2] AHMED SEGHIR Zianou ,thèse de doctorat « Evaluation de la qualité »,24/06/2012 ,pp39
- [3] Fangfang Guo, « A Contrast Enhancement Framework with JPEG Artifacts Suppression»,National University of Singapore.
- [4] S. Elkosantini, Introduction à la logique floue : les concepts fondamentaux et applications.
- [5] ARIA NOSRATINI, «Enhancement of JPEG-Compressed Images by Re-application of JPEG», Department of Electrical Engineering, University of Texas at Dallas, Received August 2, 1999; Revised April 13, 2000, Journal of VLSI Signal Processing 27, 69–79, 2001.
- [6] Yu Li¹, Fangfang Guo¹, Robby T. Tan², and Michael S. Brown¹ National University of Singapore« A Contrast Enhancement Framework with JPEG Artifacts Suppression »,2001
- [7] Frédérique Crété-Roffet. Estimer, mesurer et corriger les artefacts de compression pour la télévision numérique. Traitement du signal et de l'image [eess.SP]. Université Joseph-Fourier - Grenoble I, 2007.
- [8] Bourada Sara ,mémoire de master « Filtrage Automatique des images satellitaires bruitées des zones urbaines » chapitre 2 filtrage des images numérique ,pp 27