MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE
LA RECHERCHE SCIENTIFIQUE
UNIVERSITE MOULOUD MAMMERI DE TIZI - OUZOU
FACULTE DE GENIE ELECTRIQUE ET D'INFORMATIQUE
DEPARTEMENT D'AUTOMATIQUE

MEMOIRE DE FIN D'ETUDES

En vue de l'obtention du diplôme d'ingénieur d'état en automatique

THEME

Compression d'images par régions d'intérêt : application aux images médicales

Présenté par :
M^{elle} : MAMMAR Lamia
M^{elle} : YOUYOU Malha

Dirigé par :
M^{r} : AMEUR Soltane

Promotion: 2007-2008

Remerciements

On voudrez exprimer nos sincères remerciements et notre gratitude à M^r^ le professeur AMEUR Soltane de nous avoir encadré, motivées, suivi et soutenu tout au long de ce travail

On tient à exprimer notre reconnaissance à notre camarade Talem Hand pour son aide précieuse.

Un grand merci a nos camarades (du département automatique et électronique) qui nous ont aidés.

A tous nos amis(es).

Enfin, on doit tout à nos chers parents et familles respectives, qui nous ont soutenus et encouragées.

Bisous à zimou.

Sommaire

SOMMAIRE

Introduction générale …… 7

Chapitre I Généralités sur le traitement d'images

I-1-Introduction…….. 9
I-2-Définition de l'image…… 9
I-3-Définition d'une image numérique……... 9
I-4-Caractéristiques d'une image numérique…… ……... 10
I-4-1-Pixel…….. 10
I-4-2-Contour……. 10
I-4-3-Résolution……. 10
I-4-4-Dimension……. 11
I-4-5-Luminance……. 11
I-4-6-Contraste……... 11
I-4-7-Histogramme……. ... 12
I-4-8-Bruit…… . 14
I-5-Images binaires……. 14
I-6-Images en niveau de gris…….. 15
I-7-Images en couleurs……. 16
I-8-Qualitéde l'image numérique……... 16
I-9-Les systèmes numériques de traitement d'images……... 17
I-9-1-Prétraitement……. 18
I-9-2-Traitement des images…….. 18
I-9-3-Post-traitement…….. 18
I-10-Amélioration d'images……... 18
I-11-Conclusion…….. 19

Chapitre II Compression d'images numériques

II-1-Introduction…… 21
II-2-Définition de la compression……... 21
II-3-Mesure des performances…… 21
II-3-1-Taux de compression……... 21
II-3-2-Mesure de distorsion…… 22
II-3-3-L'entropie…… 23
II-4-Principe général de la compression d'images…… 24
II-4-1-Décorrélation…… 24
II-4-2-Quantification…….. 24
II-4-3-Codge……... 25
II-5-Méthodes de compression……... 25
II-6-Les méthodes sans perte (réversibles)…… 26
II-6-1-Codage de Shannon-Fano…… 26
II-6-2-Codage de Huffman…… 28
II-6-3-Codage en RLE (Run Lenght Encoding)…… 30
II-6-4-Codge arithmétique…….. 31
II-7-Méthodes avec pertes (réversibles)…… 33
II-7-1-Sous échantillonnage……... 33
II-7-2-Compression par dictionnaire…….. 34
II-7-2-a-Quantification scalaire (QS)…… 34
II-7-2-b-Quantification vectorielle (QV)…… 35
II-7-2-b1-Codeur…… 36
II-7-2-b1-Décodeur…… 36
II-7-3-Méthodes par transformée…… 37
II-7-4-Codage par DPC (Differentiel Predictive Coding)…….. 37

II-7-5-Décomposition en sous bandes…… 38
II-7-6-Codage par ondelettes…….. 38
II-7-7-Codage par fractales…… 38
II-8-Les normes de compression d'images……. 39
II-8-1-La norme de compression JPEG…….. 39
II-8-2-La norme de compression JBIG……... 40
II-8-3-La norme de compression H-261…… 41
II-9-Conclusion……... 41

Chapitre III Méthode adoptée

III-1-Introduction…… 43
III-2-Interet de codage par région d'intérêt…… 43
III-3-Méthode adoptée…… 44
III-3-1-Classification…….. 44
III-3-2-Codage des zones hétérogènes……... 45
III-3-2-a-Codge d'images avec la TCD…… 45
III-3-2-a1-La transformée en cosinus discrète (TCD)…… 46
III-3-2-a2-La quantification…… 51
III-3-2-a3-Codage……... 53
III-3-3- Codage des zones homogènes……... 55
III-3-3-a-Codage d'images par la QV…… 55
III-3-4-Somation……. 57
III-4-Conclusion …… 57

Chapitre IV Tests et résultats

IV-1-Introduction…… 59
IV-2-Applictions et résultats…… 60
IV-3-Résultats de codage d'image classifiées par TCD-QV……... 61
IV-3-Discution des résultats……. 74
IV-4-Conclusion……... 74

Conclusion générale……. 76

Annexes

Bibliographie

Introduction Générale

L'imagerie médicale est un domaine en plein essor, du fait du développement des technologies numériques; elle permet une investigation de plus en plus fine des organes humains.
Le problème crucial qui se pose pour l'utilisation de ces images est la grande quantité d'informations (de données) nécessaire à leur représentation. Pour remédier a ça la communauté scientifique a élaboré différentes méthodes de compression; celle-ci ont pour but de minimiser la quantité d'information nécessaire a la représentation d'une image, ceci en vu d'un archivage efficace de ces images et des données associées que l'on pourra consulter, analyser ou étudier éventuellement a distance.
Pour un bon nombre d'images médicales, l'information clinique peut être concentrée dans une région de l'image ; les méthodes de compression qui sont capables de restituer la meilleure qualité de reconstitution pour cette région deviennent alors très intéressantes alors que le reste de l'image peut être représenté avec une moindre qualité, ce qui pourrait fournir un codage d'images efficace et précis pour le domaine médical.

Notre travail consiste a développer une méthode de compression d'image par régions d'intérêt en utilisant la transformée cosinus discrète (TCD) et la quantification vectorielle (QV).

L'étude que nous avons effectuée est organisée en cinq chapitres :

Le premier chapitre porte sur des notions et généralités sur la compression d'images.
Le second chapitre est consacré au codage d'images par régions d'intérêt et aux méthodes de compression utilisées dans l'approche adoptée.
Au troisième chapitre, nous avons exposé la méthode adoptée.
Le quatrième porte sur des généralités sur l'imagerie médicale
Le cinquième chapitre de ce mémoire représente les résultats expérimentaux obtenus.
Enfin, nous terminerons par une conclusion générale

Chapitre I

Généralités sur le traitement d'images

I-1-Introduction

Avec la parole, l'image constitue 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 support d'information est désormais présent dans notre vie quotidienne à travers l'utilisation d'Internet et la manipulation de documents multimédia.

De ce fait, le traitement numérique de l'image est l'ensemble des méthodes et techniques permettant l'amélioration du contenu d'une image afin d'extraire les informations pertinentes selon les besoins et les domaines d'application.

I-2-Définition de l'image [1,7]

D'une manière générale, on définit une image comme étant un support d'information. Elle présente une scène qu'a été capte soit par un appareil photographique, soit par un satellite …etc.

Elle peut être décrite sous la forme d'une fonction I(x, y) de brillance analogique continue, définie dans un domaine borné, tel que x et y sont les coordonnées spatiales d'un point de l'image et I est une fonction d'intensité lumineuse et de couleur. Sous cet aspect, l'image est inexploitable par la machine, ce qui nécessite sa numérisation.

I-3-Définition d'une image numérique [1,7]

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, où calculé à partir d'une description interne de la scène à représenter.

Elle est obtenue par une conversion de celle-ci de son état analogique (distribution continue d'intensités lumineuses dans un plan xoy en une image numérique représentée par une matrice bidimensionnelle de valeurs numériques f(x, y) où :

x, y : coordonnées cartésiennes d'un point de l'image.

f(x, y) : niveau de gris en ce point

I-4-Caractéristiques d'une image numérique [7]

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

I-4-1-Pixel

Un pixel est l'abréviation de l'expression anglaise (Picture Elements) qu'est le plus petit point carré de l'image, c'est une entité calculable qui peut recevoir une structure et une quantification.

Si le bit est la plus petite unité d'information que peut traiter un ordinateur, le pixel est le plus petit élément que peuvent manipuler les matériels et logiciels d'affichage ou d'impression.

La quantité d'information que véhicule chaque pixel donne des nuances entre images monochromes et images couleurs. Dans le cas d'une image monochrome chaque pixel est codé sur un octet et la taille mémoire nécessaire pour afficher une telle image est directement liée à la taille de l'image.

Dans une image couleur (R.V.B.), un pixel peut être représenté sur trois octets : un octet pour chacune des couleurs : rouge (R), vert (V) et bleu (B).

I-4-2-Contour

Les contours représentent la frontière entre les objets de l'image, ou la limite entre deux pixels dont les niveaux de gris représentent une différence significative.

I-4-3-Résolution

La résolution d'une image numérique est la finesse (ou la clarté) des détails atteints par un moniteur ou une imprimante dans la production d'une image. Sur les moniteurs d'ordinateurs, la résolution est exprimée en nombre de pixels par unité de mesure (pouce ou centimètre). On utilise aussi le mot résolution pour designer le nombre de pixels affichable horizontalement ou verticalement sur un moniteur ; plus grand est ce nombre, meilleur est la résolution.

I-4-4-Dimension

Elle définit la taille de l'image. Cette dernière se présente sous forme de matrice dont les éléments sont des valeurs numériques représentatives des intensités lumineuses (des pixels).

Le produit entre son nombre de lignes et son nombre de colonnes nous donne le nombre total de pixels dans l'image.

I-4-5-Luminance

La luminance 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 celle-ci.
Une bonne luminance est caractérisée par :

* 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

I-4-6-Contraste

C'est l'opposition marquée entre deux régions d'une image, plus précisément entre les régions sombres et les régions claires. Le contraste est défini en fonction des luminances de deux zones d'images. Une augmentation du contraste a pour effet que les couleurs sombres deviennent encore plus sombres et les couleurs claires encore plus claires. Les différences entre les couleurs claires et les couleurs sombres sont donc accentuées. Cela se remarque tout particulièrement pour les photos par exemple aux changements de couleurs francs, aux côtés d'une maison où l'un est éclairé par le soleil et l'autre à l'ombre.

Si L1 et L2 sont les degrés de luminosité respectivement de deux zones voisines A1 et A2 d'une image, le contraste C est défini par le rapport :

$$C = \frac{L1 - L2}{L1 + L2} \qquad \textbf{(1-1)}$$

Avec un contraste fortement élevé les finesses dans les dégradés de couleurs se perdent, on en arrive à un effet d'ombre portée. Avec un contraste fortement réduit on en arrive à un effet de voile gris.

Les trois illustrations suivantes montrent au milieu la photo de base, à gauche un contraste réduit (Voile gris) et à droite un contraste augmenté (ombre portée).

Figure I-1: Exemple de contraste

I-4-7-Histogramme

L'histogramme des niveaux de gris ou des couleurs d'une image est une fonction qui donne la fréquence d'apparition de chaque niveau de gris (couleur) dans l'image. Pour diminuer l'erreur de quantification, pour comparer deux images obtenues sous des éclairages différents ou encore pour mesurer certaines propriétés sur une image, on modifie souvent l'histogramme correspondant.

Pour chaque graphique en pixels, il est possible d'établir un histogramme. C'est la représentation de la répartition de la luminosité et de la couleur dans le graphique. Un système d'axes sert à la représentation avec les abscisses (horizontales) et les ordonnées (vertical). Les abscisses expriment la luminosité. L'origine de l'axe figurant pour une obscurité maximale. Plus on avance sur l'axe, plus la luminosité est forte. Les ordonnées expriment le nombre de pixels dans l'image. Dans le système d'axes, il est maintenant possible de lire combien de pixels ont quelle valeur de couleur ou quelle luminosité.

Il permet de donner un grand nombre d'information sur la distribution des niveaux de gris (couleur) et de voir entre quelles bornes est repartie la majorité des niveaux de gris (couleur) dans les cas d'une image trop claire ou d'une image trop foncée.

Il peut être utilisé pour améliorer la qualité d'une image (Rehaussement d'image) en introduisant quelques modifications, pour pouvoir extraire les informations utiles de celle-ci

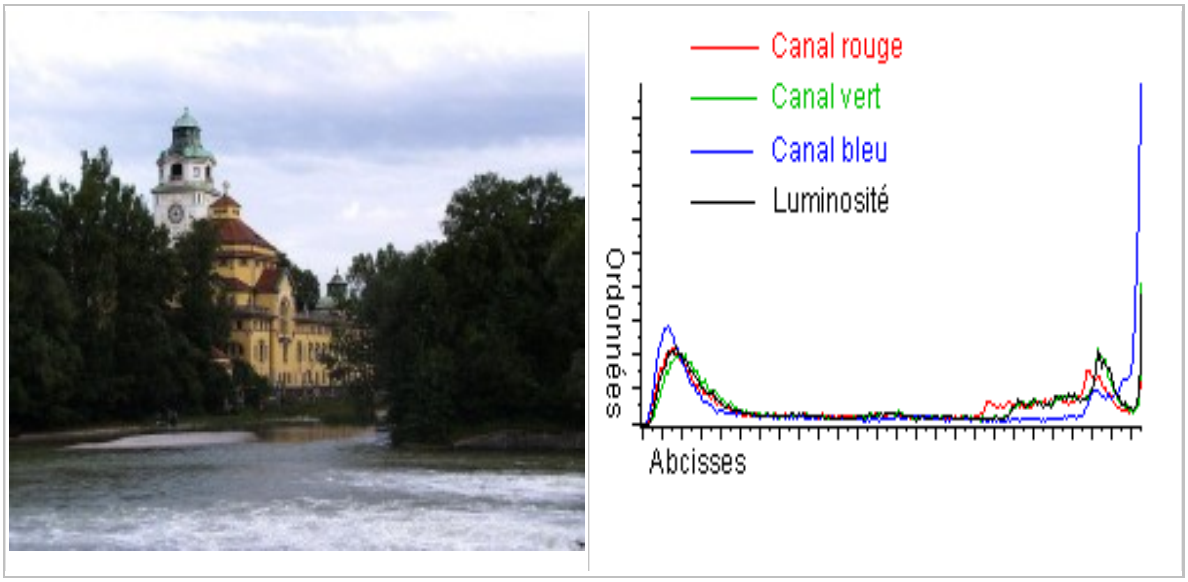


Figure I-2 : Exemple d'une photo et son histogramme

I-4-8-Bruit

Un bruit ou un parasite dans une image est considéré comme un phénomène de brusque variation de l'intensité d'un pixel par rapport a ses voisins, il provient de l'éclairage des dispositifs optique et électronique du capteur.

I-5-Images binaires

Dans une image binaire chaque pixel ne peut prendre que deux valeurs, 0 pour le noir, 255 pour le blanc. Elle contient seulement deux niveaux de gris différent numérisés en 1 bit/pixel.

Ce type de codage ne convient qu'aux images où le contraste entre les zones foncées ou claires est important. Il ne peut tenir compte des nuances de couleurs. Il est utilisé pour les images dites " au trait " (textes imprimés ou manuscrits, dessins à la plume sur fond blanc, gravures)

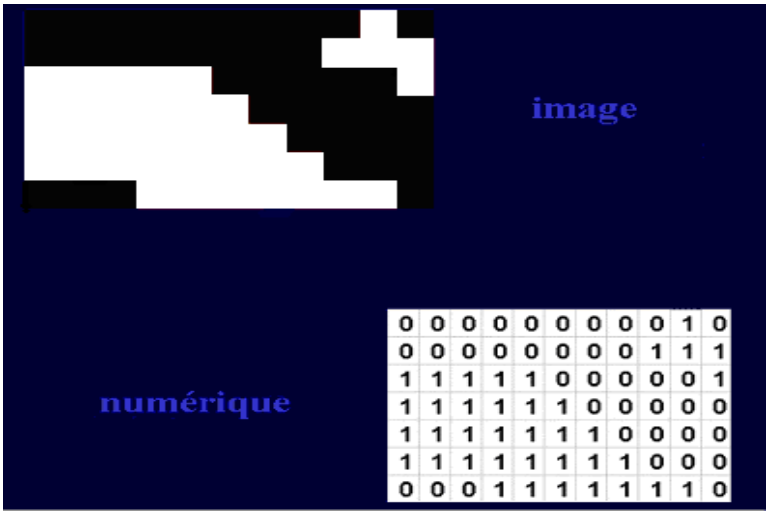


Figure I-3 : Exemple d'un détail d'une image binaire dont la couleur est codée en 0 ou en 1

I-6-Images 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, on peut attribuer à chaque pixel de l'image une valeur correspondant 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. Pour cela, il faut que le matériel utilisé pour afficher l'image soit capable de produire les différents niveaux de gris correspondants.

Le nombre de niveaux de gris dépend du nombre de bits utilisés pour décrire la " couleur " de chaque pixel de l'image. Plus ce nombre est important, plus les niveaux possibles sont nombreux.

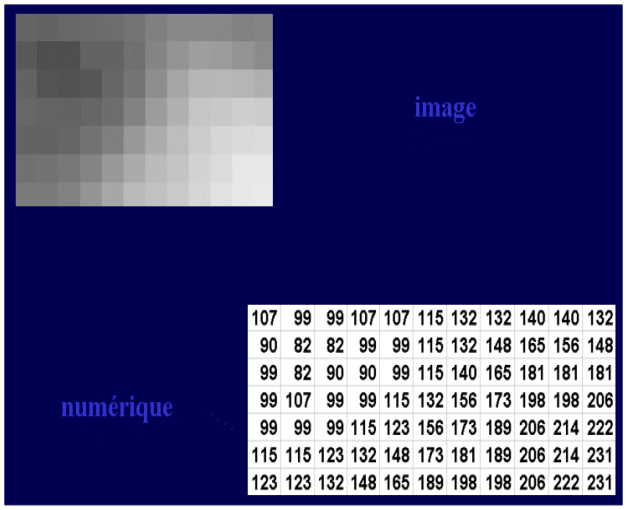


Figure I-4 : Exemple d'un détail d'une image en niveaux de gris dont la valeur de gris est codée entre 0 et 255

I-7-Images en couleurs

Au lieu de se limiter à la luminance, il faut ici mesurer et quantifier l'ensemble des informations de couleur pour chaque pixel de l'image. Pour coder ces informations, on se fonde sur un système de représentation des couleurs. On peut représenter les couleurs à l'aide de leurs composantes primaires. Les systèmes émettant de la lumière (écrans d'ordinateurs,...) sont basés sur le principe de la synthèse additive : les couleurs sont composées d'un mélange de rouge, vert et bleu (modèle R.V.B.).

L'intensité lumineuse correspondant à chacune de ces trois couleurs est numérisée comme dans le cas des images monochromes. On constate de même que 256 niveaux suffisent pour chaque couleur, le codage pouvant se faire sur 8 bits par couleur de base. On parle alors de *codage sur 24 bits en RVB*.

I-8-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 :

* La qualité des périphériques de numérisation de l'image, du nombre de niveaux de gris ou de couleurs enregistrées, …etc.

*La qualité de l'affichage à l'écran : définition de l'écran, nombre de teintes disponibles simultanément, calibrage de l'écran, …etc.

I-9-Les systèmes numériques de traitement d'images

Un système numérique de traitement d'image comprend un ensemble de technologies et de services. Les fonctions élémentaires sont par exemple l'acquisition d'images ou de données, la numérisation, la compression, le stockage, la manipulation de données, l'affichage et la transmission ou encore la manipulation a distance d'instruments

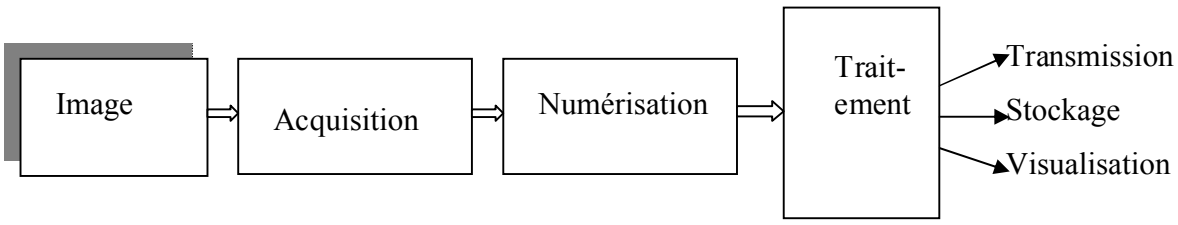


Figure I-5 : Système numérique de traitement d'image

I-9-1- Prétraitement

L'acquisition des images est la première étape dans une chaîne de traitement numérique des images, pour pouvoir manipuler une image sur un système informatique ; il est donc nécessaire de subir à l'image une transformation qui la rendra lisible et manipulable par ce système .Cette étape est appelée numérisation.

I-9-2-Traitement des images

Dans cette étape de la chaîne de traitement, nous allons faire aux images plusieurs opérations : filtrage numérique, détection de contours, calcul gradient, la transformée de fourrier ou la décomposition en ondelettes, que nous allons présenter en détail dans les prochains points.

I-9-3- Post-Traitement

Le post-Traitement concerne les images traitées ; c'est la dernière étape dans un système numérique de traitement d'images. Les images traitées seront transmises, stockées ou visualisées.

I-10-Amélioration d'images

Les images brutes permettent rarement de parvenir à une extraction directe des objets à analyser :

- soit parce que l'éclairement de l'objet n'est pas uniforme ;
- soit parce que l'objet est perçu à travers un bruit assez important : les images contiennent donc un signal (ce qui nous intéresse) et du bruit (dont on veut éliminer la plus grande partie possible) ;
- soit encore parce que le contraste n'est pas suffisant.

Avant d'extraire les objets et d'analyser une image, il va donc être nécessaire d'améliorer l'image.

-En améliorant le rapport signal sur bruit

Le transfert de l'image d'un objet jusqu'à l'ordinateur se produit avec un certain bruit. Le bruit est dû en particulier aux imperfections de la **source** qui génère l'image, au capteur proprement dit (caméra, détecteur électronique ou X) et enfin à toute l'électronique de transfert.

On améliorera la qualité de l'image en modifiant le rapport signal sur bruit. La méthode la plus simple consiste à appliquer le principe des analyseurs multicanaux, c'est à dire à effectuer plusieurs acquisitions de l'image : plusieurs sommations du signal. Le bruit n'apparaissant statistiquement jamais au même point, il sera uniformément réparti, alors que le signal, apparaissant toujours au même endroit, sera amplifié. Cette opération élimine une bonne partie du bruit (en effet, supprimer tout le bruit est quasiment impossible).

-Par filtrage

L'amélioration du rapport signal sur bruit, bien qu'intéressante, ne suffit pas toujours pour obtenir de bonnes images.

L'amélioration de l'image est essentiellement obtenue par ce que l'on appelle une opération de "Filtrage». Il existe un grand nombre de filtres possibles, mais à quelques exceptions près on peut les classer en 2 grandes catégories :

- les filtres linéaires.
- les filtres morphologiques.

L'utilisation des **filtres linéaires** provient de l'extension des méthodes mises au point pour le traitement du signal, au traitement des images numériques on peut, en effet considérer une image numérique comme un signal bidimensionnel numérisé.
Les **filtres morphologiques**, par contre, ont été conçus directement en vue d'être appliqués sur une image numérique

I-11-Conclusion

Ce chapitre est consacré à la représentation des notions générales portant sur l'image et ses représentations ; ce sont des notions de base qui nous permettent de s'introduire au domaine du traitement d'images.

Chapitre II

Compression d'images numériques

II-1-Introduction

Vue l'utilisation massive de l'imagerie numérique, la compression devient une nécessité afin d'assurer l'archivage d'une part et faciliter la transmission d'autre part.
La compression doit permettre de minimiser l'information nécessaire à la représentation des images.

II-2-Définition de la compression

La compression d'images est la réduction du nombre de bits par pixel à stocker ou à transmettre, en exploitant la redondance informationnelle dans l'image et/ou l'extraction de l'information pertinente.
Le but de la compression d'images est de modifier la représentation initiale des données pour qu'elles occupent moins de place. Cette nouvelle représentation sera décodée durant une procédure de décompression pour reconstruire l'image.

II-3-Mesure des performances

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

II-3-1-Taux de compression

Il sert à mesurer l'efficacité d'une méthode de compression.

On appel Rc : rapport de compression

$$Rc = \frac{Nombre\ de\ bits\ utilisés\ \ par\ l'image\ originale}{Nombre\ de\ bits\ utilisés\ \ par\ l'image\ compressées}$$

On définit le taux de compression :

$$TC = (1 - \frac{1}{Rc}) * 100 \qquad \textbf{(2-2)}$$

L'objectif de la compression d'images est donc d'avoir un taux de compression le plus élevé possible ; toutefois la théorie de l'information donne une limite théorique au taux maximale qu'il est possible d'atteindre sans distorsion pour toute méthode de compression sur une image donnée.

II-3-2-Mesure de distorsion

La mesure de distorsion utilisée généralement en compression d'images est l'erreur quadratique moyenne MSE (Mean Square Error). Cette grandeur représente la moyenne des écarts aux carrés entre le pixel I (i, j) de l'image originale I (i, j) et celui de l'image reconstruite I' (i, j)

$$MSE = \frac{1}{M * N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} [(I(i, j) - I'(i, j)]^2 \qquad \textbf{(2-3)}$$

M : le nombre de lignes de la matrice image
N : le nombre de colonnes de la matrice image

Le rapport signal sur bruit crête pour une image est donné par l'équation suivante :

$$PSNR = 10 \log_{10} \frac{N^2 dg \max}{MSE} \qquad \text{En db} \qquad \textbf{(2-4)}$$

Avec

N_{dgmax} : le niveau de gris maximum; et l'exemple couramment utilisé en télévision numérique est N_{dgmax}=255 (dynamique de l'image)

Lorsque la reconstruction est parfaite :

$MSE \to 0$
$PSNR \to \infty$

Ces mesures sont très utilisées car elles sont très simples à calculer, cependant elles ne rendent pas compte de la qualité visuelle engendrée par la compression.

II-3-3-L'entropie

L'entropie est une mesure de la quantité d'informations que contient une image ; de ce fait une image dont tous les pixels ont la même valeur contient très peu d'informations car elle est extrêmement redondante, son entropie est faible. Par contre une image dont tous les pixels ont une valeur aléatoire, contient beaucoup d'informations et son entropie est élevée.
En pratique, le calcul de l'entropie 'H' d'une image numérique nécessite celui des probabilités P (k) des niveaux du gris.

On défini l'entropie d'un point particulier P d'une image par :

$$H = \sum_{i=1}^{m} P(n_i) * I(n_i) \qquad \textbf{(2-5)}$$

Avec :

n_i : les niveaux de gris que peut revêtir le point P
m : le nombre total de n_i
P (n_i) : la probabilité d'apparition du niveaux de gris n_i
I (n_i) : l'information propre du niveaux de gris n_i qui est définit par :

$$I(n_i) = \log_2 \frac{1}{P(n_i)} \qquad \textbf{(2-6)}$$

D'où l'entropie :

$$H(P) = -\sum_{i=1}^{m} P(n_i) * \log_2 P(n_i) \qquad \textbf{(bpp)} \qquad \textbf{(2-7)}$$

II-4-Principe général de la compression d'images

Le schéma fonctionnel de la compression est représenté dans la figure ci-dessus

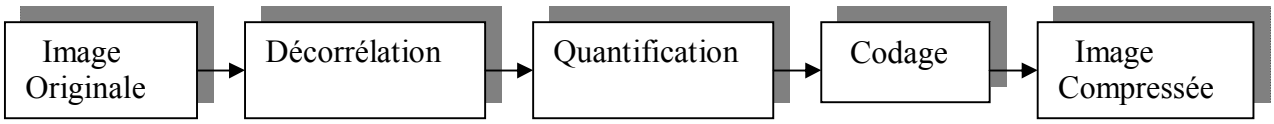


Figure II-1: Schéma général de compression

II-4-1-Décorrélation

La dépendance existante entre chaque pixel et ses voisins (la luminosité varie très peu d'un pixel à un pixel voisin) traduit une corrélation très forte sur l'image.
On essaie donc de tirer une partie de cette corrélation, pour réduire le volume d'information en effectuant une opération de décorrélation des pixels. La décorrélation consiste à transformer les pixels initiaux en un ensemble de coefficients moins corrélés, c'est une opération réversible.

II-4-2-Quantification

La quantification des coefficients obtenus 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, appelé Quantum élémentaire ou Pas de quantification. Elle peut être scalaire ou vectorielle.
Un des résultats fondamentaux des travaux de Shannon concernent la relation débit /distorsion, montre que l'on obtient de meilleures performance en utilisant la quantification vectorielle.

II-4-3-Codage

Une fois les coefficients quantifiés, ils sont codés, un codeur doit satisfaire à priori les conditions suivantes :

-unicité : deux messages différents ne doivent pas être codés de la même façon ;

-déchiffrabilité : deux mots de codes successifs doivent être distingués sans ambiguïté.

II-5-Méthodes de compression

La compression de l'image fait appel à une variété d'algorithmes du codage qui exploitent les différents types de redondances existantes dans celle-ci. Le choix et l'association de ces algorithmes se fait en fonction des applications visées et des débits souhaités.

On distingue deux grandes catégories d'algorithmes :

-Sans perte (réversibles).

-**A**vec perte (irréversibles).

Les expérimentations menées montrent que généralement les méthodes qui atteignent des taux de compression très élevés sont des méthodes avec distorsion par contre les méthodes sans distorsion engendrent des taux de compression très faibles et ne sont utilisées que dans les applications sensibles telles que les images médicales et les images satellites.

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

-la qualité de reconstitution de l'image ;

-le taux de compression ;

-le temps d'exécution ;

-la complexité de la réalisation.

II-6-Les méthodes sans pertes (réversibles)

La compression sans pertes est basée sur la théorie de l'information développée par Shannon]. Le principe de cette compression est d'éliminer la redondance présente dans les données à transmettre afin de maximiser la quantité moyenne d'information par bit de données transmis, sans perdre aucune information par rapport aux données initiales.
Les différents algorithmes de compression sans pertes permettent de compresser un fichier sans aucune perte d'information, ils sont efficaces sur certains types de données comme les textes mais ils ne permettent qu'un taux de compression relativement faible dans le cas des images.
En effet l'efficacité de la compression dépend de la redondance présente dans les données à coder, or les images ont en général une redondance très faible. Ceci justifie le développement d'autres méthodes de compression (sans pertes).

Nous en présentons ici quelques unes :

II-6-1-codage de Shannon-Fano [8, 10,13]

Ce procédé construit un arbre descendant à partir de la racine par divisions successives, le classement des fréquences se fait par ordre décroissant.

L'algorithme est le suivant :

1-classer les n fréquences non nulles par ordre décroissant ;

2-répartir la table des fréquences en deux sous tables de fréquences proches s1 et s2
Poursuivre jusqu'a ce que toutes les fréquences soient isolées ;

3-attribuer dans l'arborescence le bit «0» aux moitiés supérieures (par exemple s1), et le bit «1» aux moitiés inférieurs (s2), en suite, chaque sous table est divisée en deux autres sous tables et on ajoute de nouveau un «0» et un «1» selon le même principe que précédemment et ainsi de suite jusqu'à avoir des sous tables avec une seule fréquence ;

4-Affecter a chaque symbole le code binaire généré par l'arborescence.

Exemple de codage de Shannon-Fano :

Soit la source ayant un alphabet de cinq symboles avec les probabilités ci-dessous :

symbole	D	A	B	C	E
probabilité	14	7	6	5	4

Tableau II-1 : Table des fréquences

En appliquant les différentes étapes on aura :

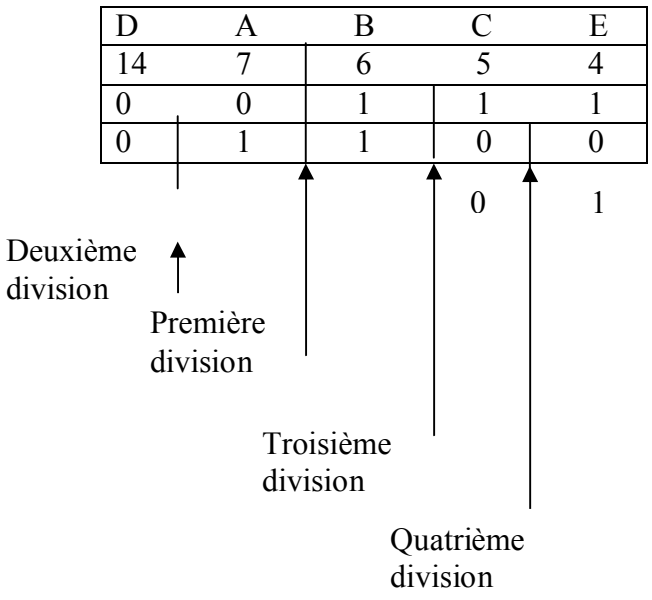


Figure II-2: Codage de Shannon-Fano

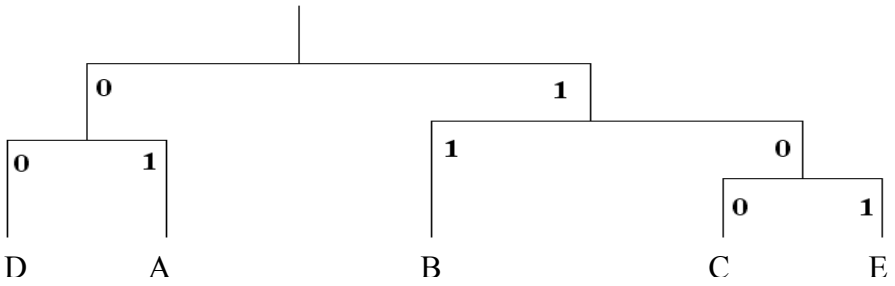


Figure II-3: L'arbre de Shannon

D'ou le codage correspondant aux différents symboles :

D	A	B	C	E
00	01	11	100	101

Tableau II-2: Les codes correspondants au codage de Shannon

II-6-2-Codage de Huffman [7,13]

Le codage de Huffman consiste à coder les symboles à transmettre, par exemple les octets d'un fichier, par des mots binaires de longueur variable suivant la probabilité d'apparition de chacun des symboles, les symboles les plus fréquents seront codés par les mots les plus courts. On code les symboles en utilisant un arbre binaire.
Il a été démontré que le codage de Huffman est optimal en ce sens qu'il minimise la taille du code par longueur entière de bits.

Le principe est le suivant :

1-répartir les fréquences f (i) des symboles ;

2-classer les symboles dans l'ordre décroissant des fréquences d'occurrences :

$$Pi = fi/\sum fi$$

Le résultat de l'algorithme ne change donc pas si l'on remplace les fréquences f_i par les probabilités ;

3-regrouper par séquences les paires de symboles de plus faible probabilité en les reclassant si nécessaire; plus précisément : calculer $S = f(in) + f(in_{+1})$;

4-choisir le plus petit indice K tel que S soit supérieure ou égal à $f(i_k)$, remplacer K par K+1;

5-recomposer la table des fréquences en plaçant à la K^{eme} position la valeur S et en décalant les autres d'une position vers le bas. Puis décrémenter n d'une unité, poursuivre jusqu'à ce que la table des fréquences ne comporte plus que deux éléments ;

6-coder avec retour arrière depuis le dernier groupe, en ajoutant un 0 ou un 1 pour différencier les symboles préalablement regroupés.

Exemple de codage de Huffman :

Soit la source ayant un alphabet de 8 symboles avec les probabilités ci-dessous :

symboles	0	1	2	3	4	5	6	7
probabilités	0.01	0.02	0.05	0.09	0.18	0.20	0.20	0.25

Tableau II-3: probabilités des symboles de la source considérée.

L'arbre de Huffman est alors représenté par la figure ci dessous :

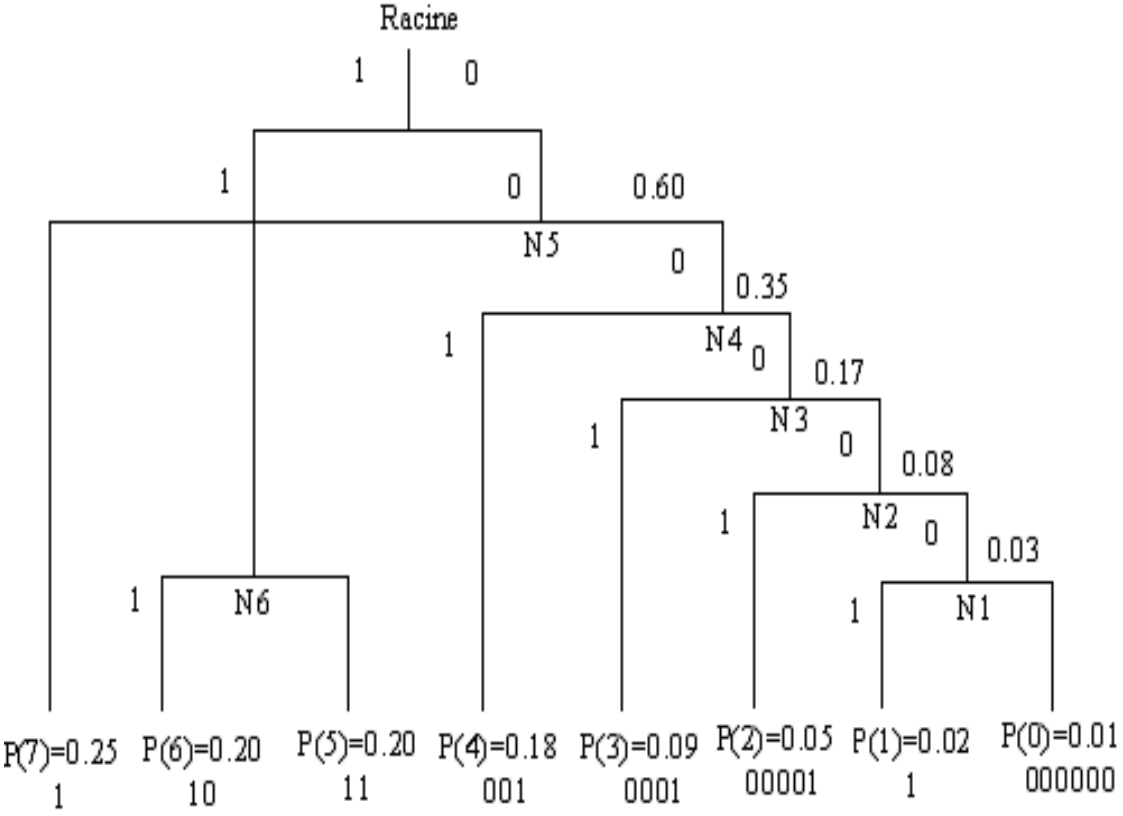


Figure II-4: Arbre de Huffman après construction

II-6-3-Codage en RLE (Run Lenght Encoding) [8]

Lorsqu'on considère une ligne de la matrice représentant une image numérique, plusieurs échantillons successifs sur cette ligne peuvent posséder la même valeur.

L'ensemble de ces échantillons est appelé « plage ».

Cette méthode consiste donc à décrire les suites de pixels identiques par leurs longueurs et leurs valeurs.

Cette compression est beaucoup moins efficace que d'autres méthodes cependant son avantage est d'être facile à implémenter.

Elle est utilisée par les fax et dans les méthodes par transformation comme JPEG.

Exemple de codage en RLE:

Soit une ligne quelconque d'une matrice représentant une image numérique :

100 100 100 101 102 102

Plage №1 : 100 100 100 (3,100) ;
Plage №2 : 101 (1, 101) ;
Plage №3 : 102 102 (2,102) ;

II-6-4-Codage arithmétique [13,7]

Contrairement aux algorithmes de Huffman et de Shannon-Fano qui associent à des symboles des motifs binaires dont la taille dépend de leur distribution, le codeur arithmétique traite le fichier dans son ensemble, en lui associant un unique nombre décimal rationnel compris entre 0 et 1, possède d'autant moins de chiffre après la virgule que le fichier dont il est redondant. Ces chiffres décimaux dépendent non seulement de des symboles du fichier dans l'ordre ou il apparaissent, mais aussi de leur distribution statistique.

Les étapes du codage arithmétique sont les suivantes :

1- calculer la probabilité associée à chaque symbole dans la chaîne coder ;

2- associer à chaque symboles un sous intervalle proportionnel à sa probabilité, dans l'intervalle [0, 1] (l'ordre de rangement des intervalles sera mémorisée car il est nécessaire au décodeur) ;

3- initialiser la limite inférieure de l'intervalle de travaille à la valeur 0 et la limite supérieure à la valeur 1 ;

4- tant qu'il reste un symbole dans la chaîne à coder :

-largeur = limite supérieure –limite inférieure

-limite inférieure = limite inférieure+ largeur *(limite basse du sous intervalle du symbole)

-limite supérieure = limite inférieure+largeur*(limite haute du sous intervalle de symbole)

5- la limite inférieure code la chaîne de manière unique.

Cette méthode est plus lente que le codage de Huffman mais permet un meilleur taux de compression.

Exemple de codage arithmétique :

Soit la source ayant un alphabet de quatre symboles avec les probabilités ci-dessous :

Symbole	A	B	C	D
Probabilité	0.4	0.3	0.2	0.1

Tableau II-4: Table des fréquences

L'intervalle initial est : [0,1[
Les messages sont codés de manière ordonnée en probabilités cumulées : [0, 0.4, 0.7, 0.9, 1[
Procédons au codage du message « ABCD » :

- **A** ∈ [0, 0.4 [
- L'intervalle [0.4, 0.7 [, du symbole B qui a l'échelle de [0, 0.4 [, devient :
  [0.4, 0.4+ (0.7-0.4)*0.4 [= [0.4, 0.52 [
- L'intervalle [0.7, 0.9 [, du symbole C qui a l'échelle de [0.4, 0.52 [, devient :
  [0.4+0.12*0.7, (0.4+0.12*0.9 [= [0.484, 0.508 [
- L'intervalle [0.9, 1 [, du dernier symbole D qui a l'échelle de [0.484, 0.508 [, devient :
  [0.5056, 0.508 [

D'où les intervalles correspondants :

Symbole	A	B	C	D
Limite inférieur	0	0.4	0.484	0.5056
Limite supérieur	0.4	0.52	0.508	0.508

Tableau II-5: Intervalles du codage arithmétique

II-7-Méthodes avec pertes (Irréversibles)

Ces méthodes permettent de retrouver une approximation de l'image numérique Les pertes sont généralement indécelables a l'œil nu.
Afin d'atteindre des taux de compression assez importants avec les images complexes, une compression irréversible est nécessaire ; elle fournit des compromis entre la qualité de l'image et le degré de compression, sachant que ces méthodes engendrent des pertes d'information même si elles sont indécelables a l'œil nu.

II-7-1-Sous échantillonnage (décimation) [5 ,8]

Le sous échantillonnage consiste a éliminer une moitié des données.
L'application de cette méthode sur un bloc d'image se fait de la manière suivante :
1-on néglige une ligne de la matrice sur deux.
2-on néglige une colonne de la matrice sur deux

Exemple :

Soit la matrice M de dimension 4*4 :

$$M = \begin{bmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{bmatrix}$$

Après sous échantillonnage on obtient la matrice M' :

$$M' = \begin{bmatrix} a_{00} & a_{02} \\ a_{20} & a_{22} \end{bmatrix}$$

Dans le cas de la reconstruction, l'image sera obtenue par interpolation en remplaçant chacun des pixels manquant par la moyenne des deux pixels adjacents.
Dans cette méthode la distorsion ne sera pas contrôlée.

II-7-2-Codage par dictionnaire

Les algorithmes de compression à base de dictionnaires codent les chaînes de symboles de longueurs variables comme des prototypes uniques. Ces prototypes forment un index dans un dictionnaire. On peut citer :

II-7-2-a-Quantification scalaire (QS) [5,8]

La quantification scalaire (QS) consiste en une projection d'un ensemble de valeurs possibles sur un autre de taille plus faible. Dans le cadre de la compression de l'image ce principe revint à remplacer les niveaux de gris de chaque pixel (n) de l'image originale (ou de sa transformée) par une valeur discrète (n_k) la plus proche au sens d'une règle prédéfinie.
Dans cette quantification, par exemple, le processus de quantification de l'amplitude des échantillons d'un signal analogique est comparé à un ensemble de niveaux de décision.
Si un échantillon d'amplitude est entre deux niveaux de décision, il sera quantifié á une valeur prédéfinie de cette bande de quantification. Dans le système numérique on assigne un code binaire à chaque échantillon quantifié.
L'importance de la QS, du point de vue compression, tient à plusieurs raisons, parmi elles :

*l'intérêt général de la quantification, dont l'application particulière est la conversion analogique/numérique;
*la simplicité qui permet des traitements en temps réel dans des applications difficiles ;
*la QS est généralement à la base d'autres méthodes.

La QS est une fonction en escalier, elle est définie comme étant une application Q qui associe à une variable continue « X », une variable discrète « Y », pouvant prendre un nombre faible et fini de valeurs.
La quantification scalaire est structuré comme suit :
-on fixe le nombre de niveaux de quantification N, donc on aura un ensemble fini de valeurs $y= \{y_j, j =1 \text{ à } N\}$
-on subdivise l'ensemble des valeurs de X, qui est l'ensemble des valeurs possibles aux symboles d'entrée, en N intervalles, on aura un ensemble $S= \{s_i, i=1 \text{ à } N+1\}$. L'application Q est alors définit par $Q(s_i, s_{i+1})=y_j$. Ceci revint à associer à chaque valeur x_i la valeur y_j.

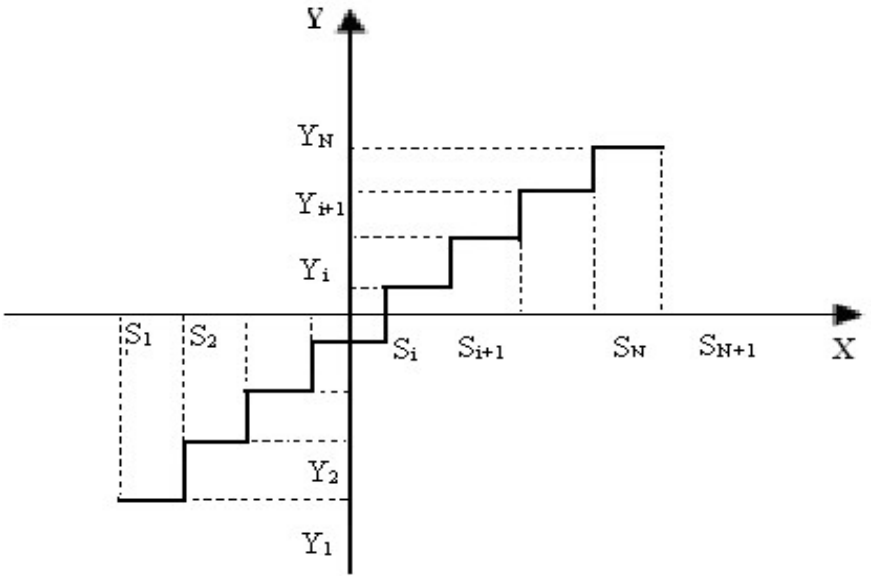


Figure II-5 : Exemple de fonction de quantification scalaire

II-7-2-b-Quantification vectorielle (QV) [5, 8,13]

Les techniques de compression d'images exploitent généralement la redondance statique présente dans l'image. La quantification scalaire associe à une variable continue une variable discrète pouvant prendre un nombre plus faible et fini de valeurs. Ces valeurs ne sont jamais totalement décorrélées, ou indépendantes. Shannon a montré qu'il était toujours possible d'améliorer la compression de données en codant des vecteurs plutôt que des scalaires.
La QV développée par Gersho et Gray a pris une place très importante dans le domaine de la compression d'images que ce soit dans le but de transmission ou celui d'archivage.

Principe de la Quantification vectorielle :

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

Elle se décompose en deux applications : codeur et décodeur.

Le Codeur :

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

Le Décodeur :

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

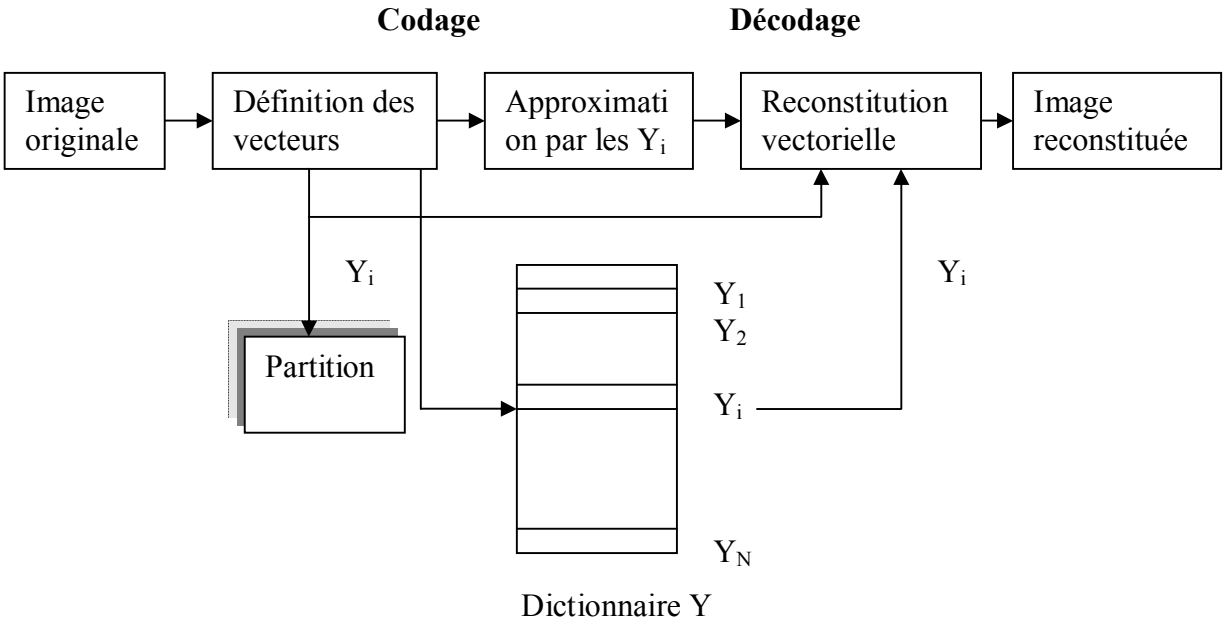


Figure II-6 : Synoptique générale du codage d'image par Quantification vectorielle

II-7-3-Méthodes par transformée [8,13]

Contrairement aux méthodes qui agissent directement sur les échantillons d'une image, les méthodes par transformation s'appliquent sur des coefficients transformés, ce sont les techniques de compression les plus utilisées grâce aux taux de compression qu'elles atteignent tout en conservant une bonne qualité de l'image.

Le principe de compression par transformation est la subdivision de l'image de dimension N x N en sous images ou blocs de taille réduite (car la quantité de calcul demandée pour effectuer la transformation de l'image entière est trop élevée).

Chaque bloc subit une transformation mathématique inversible linéaire du domaine spatial vers le domaine fréquentiel. Les coefficients obtenus indépendamment des autres blocs, sont alors quantifiés et codés en vue de leur transmission ou de leur stockage.
Pour retrouver l'intensité des pixels initiaux, on leurs applique la transformation inverse.

Parmi les transformations linéaires qui existes :

*Transformation de Karhunen-Loeve (TKL) .
*Transformation de Fourrier discrète (TFD).
*Transformation de Hadamard (TH).
*Transformation en ondelettes (TOD).
*Transformation en cosinus discrète (TCD).

II-7-4-Codage par DPC (Différentiel Prédictive Coding) [5, 7,13]

Cette méthode exploite la corrélation qui existe entre un pixel de l'image et ses voisins. Dans son principe, elle établie une fonction de prédiction qui permet d'estimer la valeur d'un pixel en fonction de son voisin (la valeur des pixels voisins). On code alors, la différence (l'erreur) de prédiction qui représente l'écart entre la vraie valeur du pixel et la valeur prédite. L'écart étant en général faible, sa représentation nécessite moins de bits que le pixel lui-même.

Les méthodes prédictives permettent une mise en œuvre facile et conduisent à des meilleurs taux de compression. Elles sont efficaces pour les images dont l'évolution temporelle ou spatiale est petite.

II-7-5-Décomposition en sous bandes [5, 7,10]

Dans un codeur en sous bandes, on commence par filtrer le signal (donc l'image) par un banc de filtres d'analyse réalisant une partition régulière des fréquences. On obtint des sous images en sous bandes, on les sous échantillonne puis on code séparément chaque signal sous échantillonné. Au codage de ses sous images, on privilégie les basses fréquences qui sont très riches en énergie, et on code plus amplement les hautes fréquences en tenant compte de la sensibilité de l'œil humain.

Au récepteur, on décode chaque composante, en réalisant un sur échantillonnage et une interpolation par un banc de filtres de synthèses ; puis on additionne les différents signaux reconstitués.
L'avantage du codage en sous bandes réside dans la possibilité d'une transmission progressive, qui permet la reconstitution de l'image compressé en basse résolution et l'ajout progressif des sous images de détails ; si on désire plus de finesse.

II-7-6-Codage par ondelettes [5,9]

La décomposition en ondelettes est une extension de la décomposition en sous- bandes. La décomposition en ondelettes aboutit à une image de détail et une image basse résolution. L'image basse résolution est a nouveau décomposée en une image de détail et une image basse résolution et ainsi de suite. Les sous images obtenues seront codées avec des stratégies différentes.

II-7-7-Codage par fractales [7,10]

Dans les techniques précedament citées, l'information pour chaque pixel est mémorisée ; par contre dans cette technique, ce sont les instructions ou des formules qui seront mémorisées pour la création de l'image.

II-8- Les normes de compression d'images [5, 7,13]

Jusqu'au début des années 80, les recherches sont essentiellement portées sur des algorithmes de compression et ont donné naissance à des normes qui permettaient des économies de 10 à 900/0, mais qui ont été très vite insuffisants devant les problèmes que posait le stockage de milliers d'images (banques des images) ou l'utilisation de séquences vidéo sur ordinateur, ce qui a rendu nécessaire la mise en place sur le plan international de groupes de coordination et d'étude , chargés de mettre ay point des standards adaptés à ces applications , afin de rendre cohérents et compatible les échanges d'informations sur les canaux de communication connus ou futurs.

II-8-1-La norme de compression JPEG

La norme JPEG (Joint Photographic Expert Group) est conçu par le groupe ISO (International Standards Organisation) et le groupe CEI (Commission Electronic International).

Elle est destinée à la compression des images fixes en couleurs et à niveaux de gris en vue de leurs stockages sur les supports numériques.

Elle a été réalisée dans la perspective de couvrir les applications les plus diversifiées en tenant compte des contraintes réalistes par rapport aux applications les plus visibles :

Publication, transmission, banque d'images.

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

-Les méthodes de compression avec pertes qui sont basées sur la TCD suivie d'une quantification et d'un codeur entropique.

-La seconde classe, concerne les processus de codage sans pertes, cette de codeur n'est pas basée sur la TCD mais sur le codage MICD suivi d'un codage entropique.

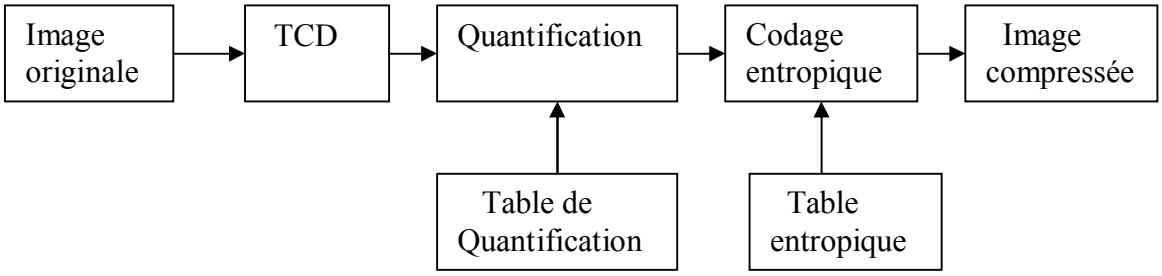


Figure II-7 : Principe de l'algorithme JPEG avec pertes

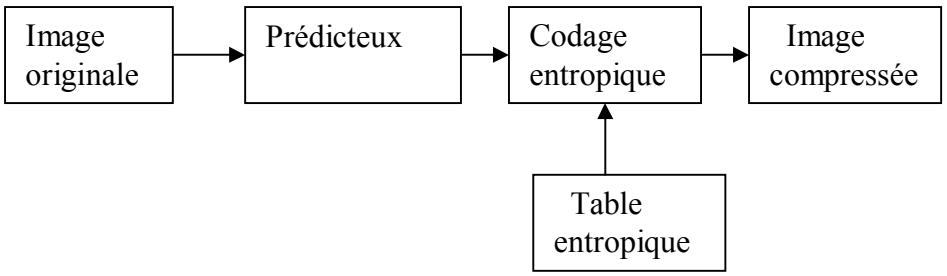


Figure II-8 : Principe de l'algorithme JPEG sans pertes

II-8-2-La norme JBIG

La norme JBIG (Joint Bi-level Image groupe) est destinée à la compression des images photographiques représentées en deux tons (noir et blanc) ainsi qu'aux images textes.
Cette norme est destinée aussi pour des débits variant de 9.6 Kbits/s a 64 Kbits/s. elle utilise un codage sans perte avec un codeur arithmétique adaptatif. Sa structure est sous forme de plusieurs couches de résolutions différentes. Cela permet une reconstruction progressive de l'image au décodage.

II-8-3-La norme de compression H-261

C'est une norme développée par C.C.I.T.T (Commission Consultative International de la Télégraphie et de la Téléphonie). Ce standard est destiné au codage des images animées pour la visiophonie (Téléphonie Visuelle).
Le H-261 utilise un codage hybride combinant la TCD et le codage prédictif.
La TCD est utilisée pour la réduction de la redondance spatial (codage intra trames).
Le codage prédictif pour réduction de la redondance temporelle entre les images de la séquence (inter- trame).

II-9-Conclusion

Nous avons vu dans ce chapitre l'utilité de la compression, qui est de réduire la quantité de mémoire nécessaire pour le stockage d'une image ou de manière équivalente de réduire le temps de transmission de celle-ci. Cette compression peut soit conserver l'image intacte, on parle alors de compression sans perte, soit autoriser une dégradation de l'image pour diminuer encore l'empreinte mémoire, on parle ici de compression avec perte.

Chapitre III

La méthode adoptée

III-1-Introduction

L'intérêt principal de la compression dans le contexte médical est la réduction de mémoire de stockage et de temps de transfert des images tout en gardant une bonne qualité est une nécessité.
Le but de notre approche est de réaliser un codage d'images numériques par régions d'intérêt, qui assure un compromis entre un taux de compression élevé et une qualité appréciable de l'image reconstruite.
Pour cela, on a fait usage des méthodes de codages par la transformée en cosinus discrète (TCD) et par la quantification vectorielle (QV). Le processus de codage consiste à extraire la région d'intérêt c'est-à-dire les informations jugées importantes dans l'image, ces derniers sont codés par une méthode qui assure le moins de pertes d'informations

.

III-2-Interet du codage par régions d'intérêt

L'atout de la compression d'images par régions d'intérêt ou ROI (Region of interest) réside dans la possibilité pour un schéma de compression de favoriser des zones visuellement sensibles de l'image tout en négligeant les parties moins significatives. Cet aspect peut être particulièrement intéressant dans les applications médicales.
Sur une image médicale, les experts cherchent des signes pathologiques sur certaines régions bien déterminées. Il est donc primordial de coder ces zones avec une méthode qui n'introduit que peu de distorsion pour une bonne qualité visuelle. En même temps, d'autres régions portent moins d'informations médicales, ce qui rend acceptable une distorsion plus élevée – permettant d'obtenir un débit plus faible – dans ces zones

Dans ce qui va suivre nous allons définir une méthode que nous avons adopté .cette dernière code les zones de l'image selon leur intérêt, elle s'intitule codage d'image classifiées.

III-3-Méthode adoptée

La méthode que nous avons élaborée est la technique de codage d'images classifiées.
Le schéma de codage/décodage adopté par cette technique est représenté dans la figure ci-dessus :

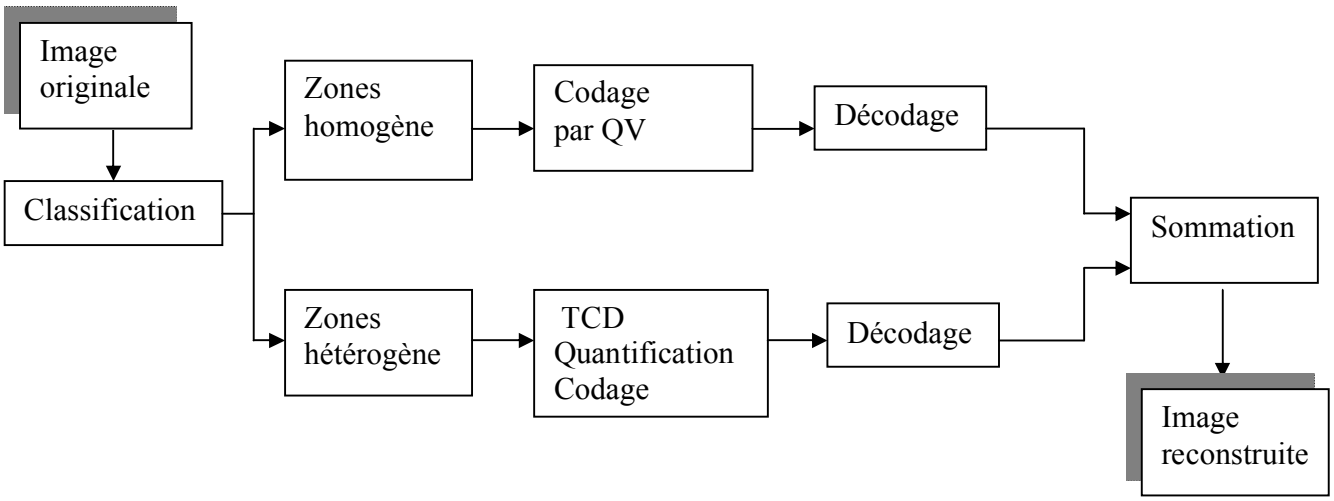


Figure III-1: Synoptique de codage/décodage d'images classifiées.

III-3-1-Classification [7]

Cette technique est basée sur la classification des blocs de l'image suivant un critère d'homogénéité en deux classes : l'une homogène, l'autre hétérogène.
Les étapes de cette technique sont données comme suit :

1-Diviser l'image en blocs de taille identique : $N=Kx \times Ky$

2-Calculer la moyenne des vecteurs des niveaux de gris ainsi que la variance pour un bloc de l'image en appliquant les formules suivantes (3-1 et 3-2).

$$\text{Moyenne : } M = \frac{1}{N}\sum_{i=0}^{Ky-1}\sum_{j=0}^{Kx-1}\text{Im}\, g(i,j) \qquad \textbf{3-1}$$

$$\text{Variance : } \delta^2 = \left(\frac{1}{N}\sum_{i=0}^{K_y-1}\sum_{j=0}^{K_x-1}\text{Im}\, g^2(i,j)\right) - M^2 \qquad \textbf{3-2}$$

Img (i, j) : valeur du niveau de gris repéré par les indices i et j

3-Fixer la valeur du seuil T dans son intervalle de variation [0.1, 1] en respectant un pas de 0.1

4-Comparer le rapport v/M avec le seuil T

-Si le rapport est inférieur ou égale a T alors la variation de l'intensité lumineuse a l'intérieur du bloc est classé comme étant homogène

-Si le rapport est supérieur à T le bloc est hétérogène

5-Répéter les étapes 2,3 et 4 pour tous les blocs qui constituent l'image

III-3-2-Codage des zones hétérogènes

On code les zones hétérogènes en utilisant la transformée en cosinus discrète

III-3-2-a-Codage d'images avec la TCD [7, 10, 13,14]

La méthode de compression par TCD est basée sur trois étapes :

1-L'application de la TCD.

2-La quantification.

3-Le codage

III-3-2-a_1-La transformée en cosinus discrète (TCD)

La transformée de cosinus discrète (TCD) est une variété de la transformée de Fourier Elle transforme un signal d'amplitude discrète bidimensionnelle (dans ce cas le niveau de gris de l'image) en une transformation bidimensionnelle de fréquences.

Pour le signal Img(x, y) les axes X et Y représentent les dimensions horizontale et verticale de l'image.

Pour la transformée de cosinus discrète du signal TCD (i, j) les axes représentent les fréquences du signal en deux dimensions.

Afin de faciliter la compréhension, décrivant la transformée de Fourrier rapide FFT (Fast Fourrier Transform) appliquée a un signal discret unidimensionnel.

Prenant pour exemple le signal temporel X (t) de la figure III-2, ce signal particulier est composé de trois courbes sinusoïdales différentes additionnées pour ne former qu'un seul signal, chaque point représente l'amplitude du signal un moment spécifique.

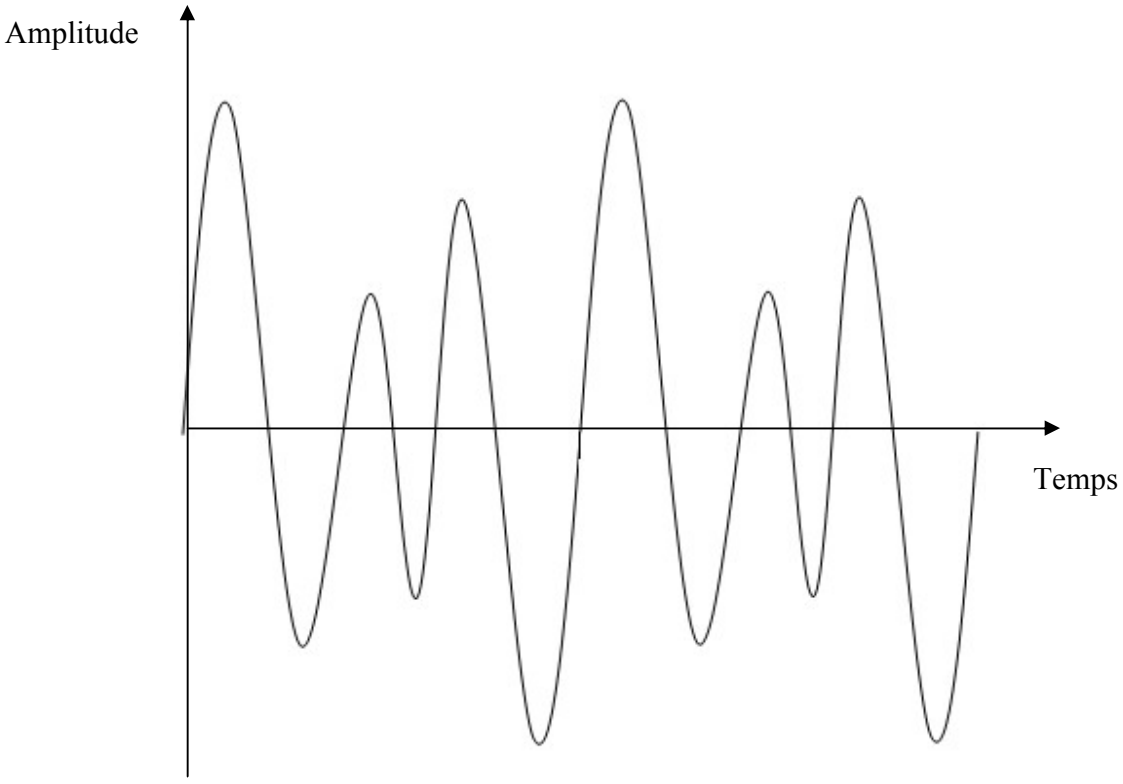


Figure III-2 : La représentation temporelle classique d'un signal

Le traitement de ce signal par la FFT donne l'ensemble de données représentées à la figure III-3
Les axes X et Y n'ont pas la même signification, chaque point de l'axes X représente sa « grandeur » dans le signal original. Dans ce cas particulier de l'addition de trois sinusoïdales, la FFT fournit les trois fréquences des sinusoïdales ainsi que leurs grandeurs. Ces six informations sont suffisantes pour la reconstruction du signal d'entrée, il sera de type :

$$Y=a_1 \sin(f_1 x) + a_2 \sin(f_2 x) + a_3 \sin(f_3 x) \quad \textbf{(3-1)}$$

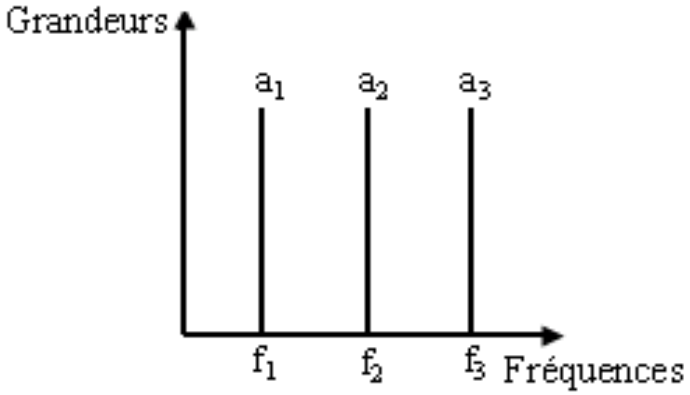


Figure III-3: Points de données après traitement par la FFT

La transformation FFT est réversible elle possède une FFT inverse qui permet de retrouver le signal d'origine. De plus les deux transformations FFT et FFT inverse sont conservatives excepté pour les pertes de précisions résultant des erreurs d'arrondi et de troncation.
La TCD travaille de la même manière que la FFT mais sur un domaine de variations bidimensionnel ; ainsi le signal Img(x, y) est transformé en une information spectrale (de fréquence) TCD (i, j). L'inverse de l TCD existe on l'appelle TCD inverse (TCDI).
La TCD s'applique sur un signal représenté par une matrice carrée, ce qui fournit une matrice carrée pour la TCDI.

Leurs formules sont les suivantes :

$$TDC(i,j) = \frac{1}{\sqrt{2N}} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} C(i)C(j)\ \mathrm{Im}\, g(x,y) \cos\left(\frac{2x+1}{2N} i\pi\right) \cos\left(\frac{2y+1}{2N} j\pi\right) \quad \textbf{(3-2)}$$

$$TDCI(x,y) = \frac{1}{\sqrt{2N}} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} C(i)C(j)\ TCD(i,j)\ \cos\left(\frac{2x+1}{2N} i\pi\right) \cos\left(\frac{2y+1}{2N} j\pi\right) \quad \textbf{(3-3)}$$

Avec :

$$C(i) = \begin{cases} \frac{1}{\sqrt{2}} & si\ i = 0 \\ 1 & si\ i > 0 \end{cases}$$

Img(x, y) : matrice carrée de dimensions N.

Img(x, y) : TCDI(x, y).

$0 \le x \qquad y \le N$.

Ces formules montrent que pour calculer la TCD en un point, il est indispensable de parcourir toutes les valeurs de Img(x, y).

La double somme nous oblige, d'un point de vue algorithmique, a effectuer N*N calculs de termes ; il est de même pour la TCDI.

Pour une image de 256x256 pixels, le calcul de la TCD nécessitera en chaque point le calcul de 65536 termes, soit en 4294967296 calculs de termes ; les calcules de l TCD et TCDI deviennent donc très vites irréalisables lorsque la taille de l'image augmente.

Pour contourner ce problème la mise en œuvre de la TCD décompose l'image en blocs plus petits et plus utilisables.

La figure suivante représente la transformée TCD sur l'image Tête :

Figure III-4: Image Tête

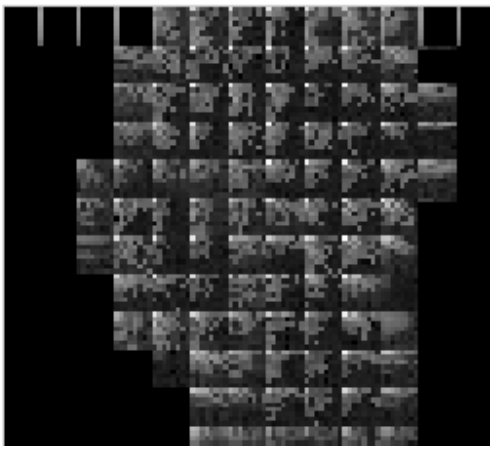


Figure III-4 : Image TCD de l'image Tête

Pour illustrer la TCD, voici un exemple qui représente un bloc de 8x8 pixels :

$$\begin{bmatrix} 139 & 144 & 149 & 153 & 155 & 155 & 155 & 155 \\ 144 & 151 & 153 & 156 & 159 & 156 & 156 & 156 \\ 150 & 155 & 160 & 163 & 158 & 156 & 156 & 156 \\ 159 & 161 & 162 & 160 & 160 & 159 & 159 & 159 \\ 159 & 160 & 161 & 162 & 162 & 155 & 155 & 155 \\ 161 & 161 & 161 & 161 & 160 & 154 & 157 & 157 \\ 162 & 162 & 161 & 163 & 162 & 157 & 157 & 157 \\ 162 & 162 & 161 & 161 & 163 & 158 & 158 & 158 \end{bmatrix}$$

Matrice III-1 : Matrice de pixels d'entrée

$$\begin{bmatrix} 235 & -1 & -12 & -5 & 2 & -2 & -3 & 1 \\ -23 & -18 & -6 & -3 & -3 & 0 & 0 & -1 \\ -11 & -9 & -2 & 2 & 0 & -1 & -1 & 0 \\ -7 & -2 & 0 & 2 & 1 & 0 & 0 & 0 \\ -1 & -1 & 2 & 2 & 0 & -1 & 1 & 1 \\ 2 & 0 & 2 & 0 & -1 & 2 & 1 & -1 \\ -1 & 0 & 0 & -1 & 0 & 2 & 1 & -1 \\ -3 & 2 & -4 & -2 & 2 & 1 & -1 & 0 \end{bmatrix}$$

Matrice III-2 : Matrice de la TCD

Après transformation, on constate que lorsque nous nous éloignons des composantes continues de l'image, nous trouvant que les coefficients deviennent moins importants pour la description de l'image.

Chaque bloc est composé de 8x8 coefficients ; le coefficient (0,0) est noté DC^1, il présente la moyenne des intensités du bloc le reste des coefficients sont notés AC^2, ils sont localisés en hautes fréquences et sont assimilés du bruit.

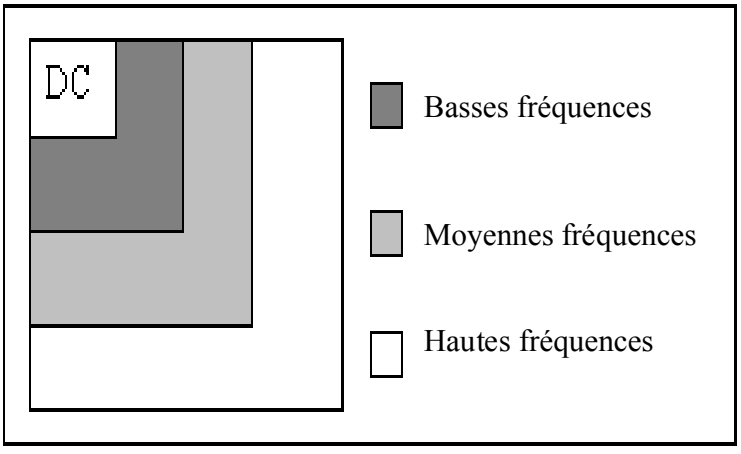


Figure III-5 : Distribution fréquentielle des coefficients TCD

III-3-2-a_2-La quantification

Le but de cette étape est de diminuer la précision du stockage des entiers de la matri c e TCD pour réduire le nombre de bits occupé par chaque entier.
C'est la seule partie non conservative de la méthode (excepté les arrondis effectués).

Puisque les informations de basses fréquences sont plus pertinentes que celle des hautes fréquences, la diminution de précision doit être plus forte dans les hautes fréquences .la perte de précision va donc être de plus en plus grande lorsqu'on s'éloigne de la position (0,0). Pour cela on utilise une matrice de quantification contenant des entiers par les quels seront divisées les valeurs de la matrice TCD. Ces entiers seront de plus en plus grands lorsqu'on s'éloigne de la position (0,0).

Les valeurs de la matrice TCD ont été arrondies à l'entier le plus proche selon la formule suivante :

$$\overline{TCD}(i,j) = \text{int}\left[\frac{TCD(i,j)}{Q(i,j)}\right] \qquad (3\text{-}4)$$

$\overline{TCD}$ (i, j) : Le coefficient TCD normalisé.
Int (x) : La fonction d'arrondis a l'entier le plus proche.
TCD (i, j) : Le coefficient TCD de position (i, j) du bloc.
Q (i, j) : La matrice de normalisation.

Les matrices de quantification intéressantes sont celles permettant de « choisir la pertes de qualité acceptable. Ce choix a été rendu possible grâce aux testes intensifs des matrices.
Habituellement, on prend pour matrice de quantification :

$$Q(i,j) = 1 + NQ\,(1+i+j) \qquad (3\text{-}5)$$

i: L'indice de ligne.
J: L'indice de colonne.
NQ : Pas de quantification (choisi entre 1 et 25).

On remarque que beaucoup de composantes de hautes fréquences de la matrice quantifiée sont tronquées a zéro, éliminant leurs effets sur l'image.
Par contre, les composantes pertinentes ont été peu modifiées.

Voici un exemple de quantification pour NQ=2

$$\begin{bmatrix} 3 & 5 & 7 & 9 & 11 & 13 & 15 & 17 \\ 5 & 7 & 9 & 11 & 13 & 15 & 17 & 19 \\ 7 & 9 & 11 & 13 & 15 & 17 & 19 & 21 \\ 9 & 11 & 13 & 15 & 17 & 19 & 21 & 23 \\ 11 & 13 & 15 & 17 & 19 & 21 & 23 & 25 \\ 13 & 15 & 17 & 19 & 21 & 23 & 25 & 27 \\ 15 & 17 & 19 & 21 & 23 & 25 & 27 & 29 \\ 17 & 19 & 21 & 23 & 25 & 27 & 29 & 31 \end{bmatrix}$$

Matrice III-3 : Matrice de quantification

$$\begin{bmatrix} 15 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ -2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Matrice III-4 : Matrice de TCD quantifiée

III-3-2-a_3-Codage

Le codage de la matrice TCD quantifiée se fait en parcourant les éléments dans l'ordre imposé par une séquence particulière appelée séquence zigzag.

On lit les valeurs en zigzag inclinées a 45° en commencent par le coin supérieur gauche et en finissant en bas droite.

Cette séquence a la propriété de parcourir les éléments en commençant par les basses fréquences et de traiter les fréquences de plus en plus hautes.

Puisque la matrice TCD contient beaucoup de composantes de hautes fréquences nulles, l'ordre de la séquence zigzag va engendrer de longues suites de 0 consécutives.

Le résultat est une suite monodimensionnelle des coefficients quantifiés numérotés de 1 à 64 pour les blocs de 8x8 (par exemple).

Ceci facilite de nouveau les étapes de compression par le codage PLC (Run Length Coding).

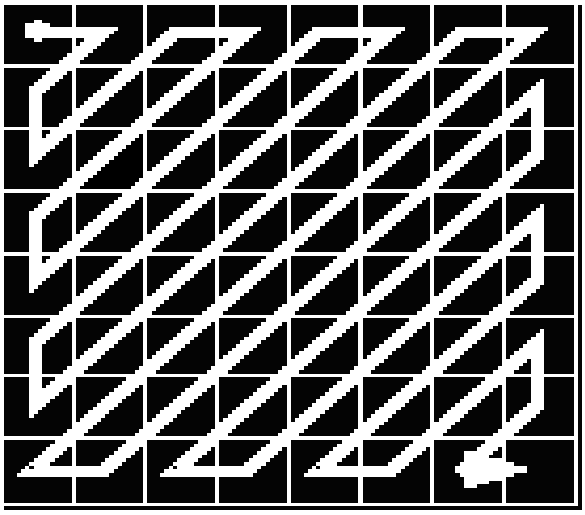


Figure III-6: Séquence zigzag de la lecture d'un bloc de 8x8 pixels

Le chemin réel de la séquence zigzag se présente comme ceci :

(0, 0),
(0, 1), (1, 0),
(2, 0), (1, 1), (0, 2),
(0, 3), (1, 2), (2, 1), (3, 0),
(4, 0), (3, 1), (2, 2), (3, 2), (0, 4),
(0, 5), (1, 4), (2, 3), (3, 2), (4, 1), (5, 0),
(6, 0), (5, 1), (4, 1), (3, 3), (2, 4), (1, 5), (0, 6),
(0, 7), (1, 6), (2, 5), (3, 4), (4, 3), (5, 2), (6, 1), (7, 0),
(7, 1), (6, 2), (5, 3), (4, 4), (3, 5), (2, 6), (1, 7),
(2, 7), (3, 6), (4, 5), (5, 4), (6, 3), (7, 2),
(7, 3), (6, 4), (5, 5), (4, 6), (3, 7),
(4, 7), (5, 6), (6, 5), (7, 4),
(7, 5), (6, 6), (5, 7),
(6, 7), (7, 6),
(7, 7),

Le réarrangement zigzag dans l'exemple suscité précédemment aboutit au vecteur V suivant : V= [15, 0, -2, -1, -1, …..].

Les suite de valeurs nulles sont simplement codées en donnant le nombre de 0 successifs (codage RLE).

III-3-2-Codage des zones homogènes

On code les zones homogènes en utilisant la quantification vectorielle (QV)

III-3-2-a-Codage d'images par la quantification vectorielle [11]

La quantification vectorielle est basée sur l'utilisation d'un dictionnaire .le dictionnaire contient un nombre de blocs typiques .Pour chaque bloc, on recherche dans le dictionnaire le bloc le plus proche (c'est-à-dire celui qui donne la distance euclidienne la plus faible), et on transmet uniquement le numéro de ce bloc. Le récepteur qui possède le même dictionnaire que l'émetteur, va chercher dans le dictionnaire les blocs dont le numéro lui est indiqué pour l'incruster dans l'image.

La figure III-7 illustre le principe de cette méthode.

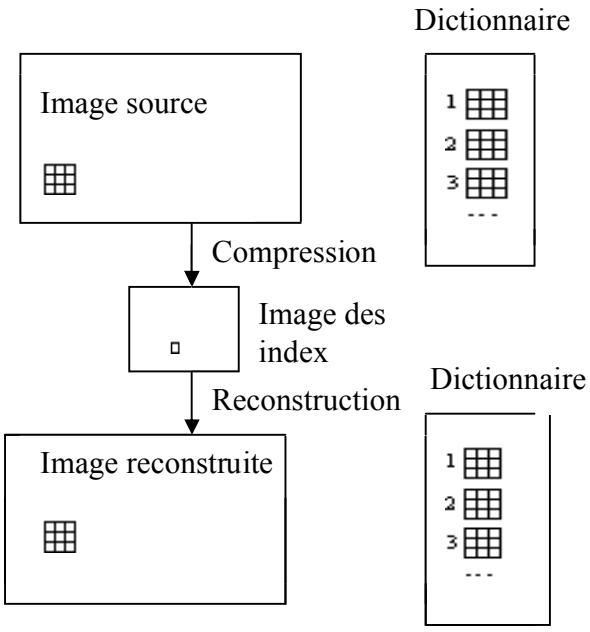


Figure III-7: Principe de la compression par quantification vectorielle

Si l'on note D la taille du dictionnaire, il faut $\log_2 D$ bits pour coder le numéro du bloc. Si l'on suppose qu'un pixel de l'image est initialement codé sur 8 bits, le taux de compression est :

$$TC = \frac{8n}{\log_2 D} \qquad (3\text{-}6)$$

Pour crée un dictionnaire efficace on utilise l'algorithme ci-dessous (on fixe un nombre d'itérations, en fonction du temps de calcul disponible et des performances souhaitées).Cet algorithme est généralement appelé k-moyennes (k-means, en anglais). Il existe des variantes améliorées, telles que l'algorithme LGB, par exemple.

1. Création d'un ensemble B contenant un grand nombre de blocs extraits d'un ou plusieurs images.

2. Initialisation du dictionnaire en prenant D blocs au hasard dans B (éviter de prendre des blocs consécutifs car ils risquent d'être trop semblables, ce qui entraînerait un manque de diversité dans le dictionnaire initial).

3. Associer à chaque élément de B une étiquette : le numéro du bloc du dictionnaire le plus proche

4. remplacer chaque bloc du dictionnaire par le barycentre des élément de B qui portent son étiquette (si aucun élément ne porte son étiquette, choisir un bloc au hasard dans B).

5. Tant que le nombre d'itération n'est pas atteint, aller en (3).

III-3-3-Sommation

Après le codage des deux zones par la quantification vectorielle (QV) et la transformée en cosinus discrète (TCD) on décode ces dernières séparément puis on les additionne pour avoir une image finale compressée.

III-4-Conclusion

Nous venons de proposer une méthode de compression d'images par régions d'intérêt. Deux types de compressions sont utilisés : une pour la compression des blocs hétérogène (TCD) et une autre pour la compression des blocs homogène (QV).
Dans le chapitre qui va suivre, nous allons appliquer cette méthode sur une image médicale obtenue par IRM.

Chapitre IV

Tests et résultats

IV-1-Introduction

Il existe des logiciels de traitement d'images dans lesquels on réalise un traitement en assemblant graphiquement des boites correspondants à des traitements élémentaires ou en utilisant des menus. Ces logiciels conviennent parfaitement lorsque on doit traiter une application simple et/ou standard. Par contre, dès que le problème à traiter s'éloigne un peu d'un problème standard, on ne peut se dispenser de comprendre le traitement d'images, et de développer ses propres traitements. Dans le travail que nous avons effectué, les algorithmes de compression TCD et QV ont été illustré grâce au langage de programmation Matlab, version 7.0.

L'utilisation de Matlab nous permet de créer nos traitements exclusivement à partir de fonctions élémentaires, ces dernières sont en grande partie, des fonctions du noyau de base de Matlab. Certains traitements spécifiques (changement de format, visualisation, opérations morphologiques, etc.) font appel à des fonctions de l'extension « traitement d'images ».

Nous présenterons dans ce chapitre les résultats aux quels, nous avons aboutis, tout en montrant la qualité des simulations obtenues sur l'image.

Pour évaluer les résultats obtenus, nous allons définir les deux critères d'évaluation des performances de toute méthode de compression, qui sont :

*le calcul du rapport signal sur bruit (PSNR) qui mesure la distorsion de l'image traitée par rapport à celle d'origine, selon l'équation suivante :

$$PSNR = 10\log_{10}\left[\frac{(255)^2}{MSE}\right] \qquad \textbf{(db)} \qquad (4\text{-}1)$$

MSE : Erreur quadratique moyenne entre l'image originale et celle reconstruite, elle présente la distorsion introduite lors du processus codage/décodage.

*Le taux de compression :

Il permettra d'évaluer les performances de la méthode utilisée en fonction du nombre de bits par pixels (bpp).

$$Tc = (1-(\text{taille finale} / \text{taille originale})) \times 100 \qquad (4\text{-}2)$$

Les paramètres qui caractérisent la TCD sont :

-Pas de quantification **NQ**.

-La taille **m** des blocs de l'image.

Le paramètre qui caractérise la QV est :

-La taille du dictionnaire **Tdico**

Nous allons aussi utiliser un paramètre qui caractérise la classification qui est le seuil de classification **T**.

Nous faisons varier au maximum les paramètres cités pour estimer le rapport entre la qualité de la restitution de l'image et le taux de compression atteint.

Le processus de la compression à été implémenté sur un micro ordinateur de 180 Mhz, possédant une mémoire centrale (RAM) de96 Mo.

IV-2-Application et résultats

Pour nos tests, nous avons utilisé une image test codée sur 8bits (256 niveaux de gris) qui est représentée dans la figure ci-dessous :

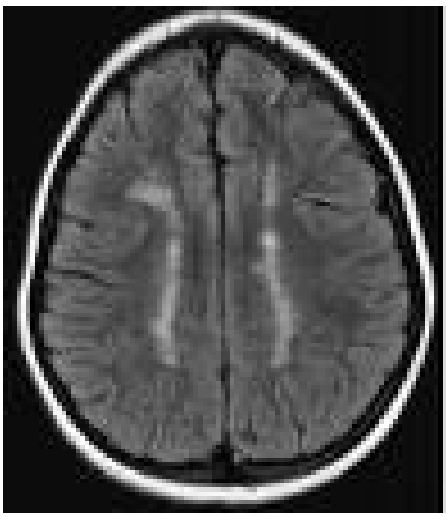


Figure IV-1: Image test

Nous avons appliqué la TCD et la QV sur l'image de la figure IV-1 et cela en procédant comme suit :

-on fixe le seuil T ce qui nous permet d'avoir deux zones de l'image : homogène et hétérogène, auxquelles on applique les méthodes de compression séparément ;

-on applique l'algorithme de compression par la TCD à la zone hétérogène ;

-on applique l'algorithme de compression par la QV à la zone homogène ;

-on fait le décodage et la reconstruction des deux zones séparément ;

-on fait la somation des deux zones décodées et reconstruites pour avoir une image finale compressée ;

-on calcul le taux de compression Tc et le PSNR de l'image finale ainsi que le temps de calcul en faisant varier les paramètres caractéristiques de la TCD et de la QV, ce qui nous permet d'avoir des tableaux qui sont établis suivant six cas.

Remarque

-on peut appliqué la TCD à la zone homogène et la QV à la zone hétérogène selon l'intérêt que nous portons à chaque zone.

-Nous avons choisie de mettre à côté de chaque tableau le graphe caractérisant la variation du PSNR en fonction du Tc ainsi que quelques exemples montrant l'image originale et l'image compressée correspondant à la variation des paramètres des méthodes utilisées.

IV-3-Résultats de codage d'images classifiées par TCD-QV

Pour mettre en évidence les performances de la méthode adoptée, nous avons effectué différents tests, les résultats obtenus sont illustrés dans les tableaux IV-1, IV-2, IV-3, IV-4, IV-5 et IV-6.

Ils représentent le PSNR, le taux de compression Tc et le temps de calcul.

Pour estimer le rapport entre la qualité de la restitution de l'image (PSNR) et le taux de compression obtenu par les différents paramètres, nous les avons représentés sous forme de graphes aux figures IV-2, IV-3, IV-4, IV-5, IV-6 et IV-7.

-Premier cas

Le tableau suivant indique la variation du taux de compression Tc et de PSNR de l'image de la figure IV-1en fonction de la taille des blocs m, en fixant le seuil T à 0,5, la taille du dictionnaire Tdico à 100 et NQ=1

m×m	PSNR (dB)	Tc (%)	Temps[H :Min:s :ms]
2×2	44,37	85,15	00 :00 :12 :81
4×4	45,46	84,20	00 :00 :12 :80
6×6	46,73	77,73	00 :00 :12 :54
8×8	46,92	48,04	00 :00 :12 :06

Tableau IV-1 : Résultats obtenus pour le premier cas

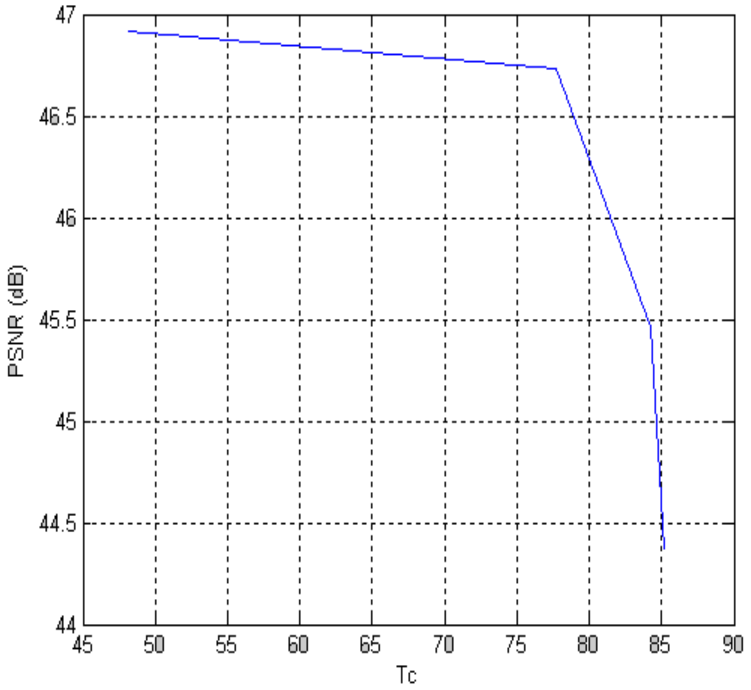
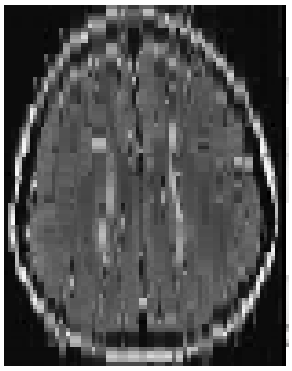
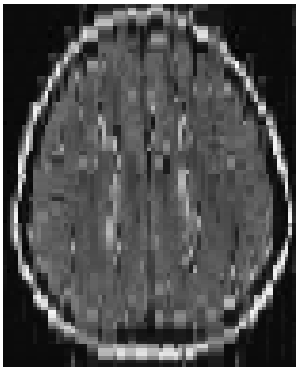


Figure IV-2 : Variation du PSNR en fonction de Tc pour le 1er cas



Tc=77.73



Tc=84.20

Figure IV-2-a: Images restituées pour le 1er cas

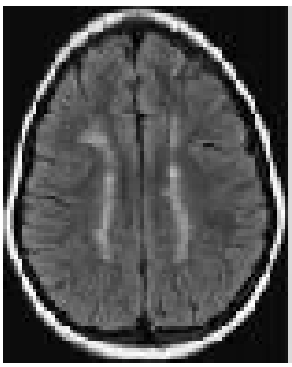


Image test

-Deuxième cas

Le tableau suivant indique la variation du taux de compression Tc et de PSNR de l'image de la figure IV-1 en fonction de la taille des blocs m, en fixant le seuil T à 0,5, la taille du dictionnaire Tdico à 100 et NQ=5

m×m	PSNR (dB)	Tc (%)	Temps[H :Min:s :ms]
2×2	30,39	84,04	00 :00 :12 :89
4×4	31,48	83,26	00 :00 :12 :86
6×6	32,75	76,95	00 :00 :12 :80
8×8	32,98	46.09	00 :00 :12 :73

Tableau IV-2 : Résultats obtenus pour le deuxième cas

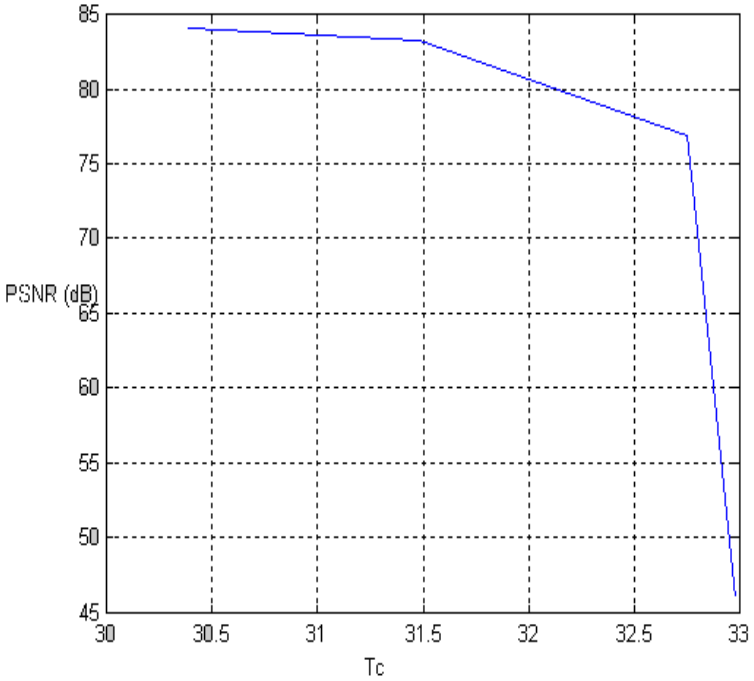
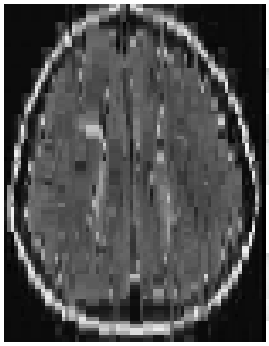
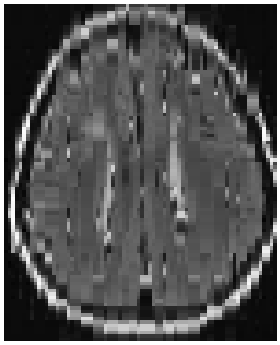


Figure IV-3: Variation du PSNR en fonction de Tc pour le 2eme cas



Tc=76,95



Tc=83,26

Figure IV-3-a: Images restituées pour le 2eme cas

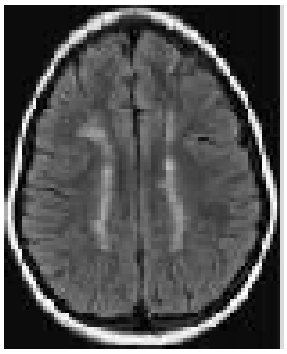


Image test

-Troisième cas

Le tableau suivant indique la variation du taux de compression Tc et de PSNR de l'image de la figure 4-1en fonction de la taille des blocs m, en fixant le seuil T à 0,4, la taille du dictionnaire Tdico à 256et NQ=1.

m×m	PSNR (dB)	Tc (%)	Temps[H :Min:s :ms]
2×2	35,11	71,56	00 :00 :13 :00
4×4	35,16	73,75	00 :00 :13 :06
6×6	36,26	62,81	00 :00 :13 :04
8×8	36,41	49,53	00 :00 :12 :05

Tableau IV-3: Résultats obtenus pour le troisième cas

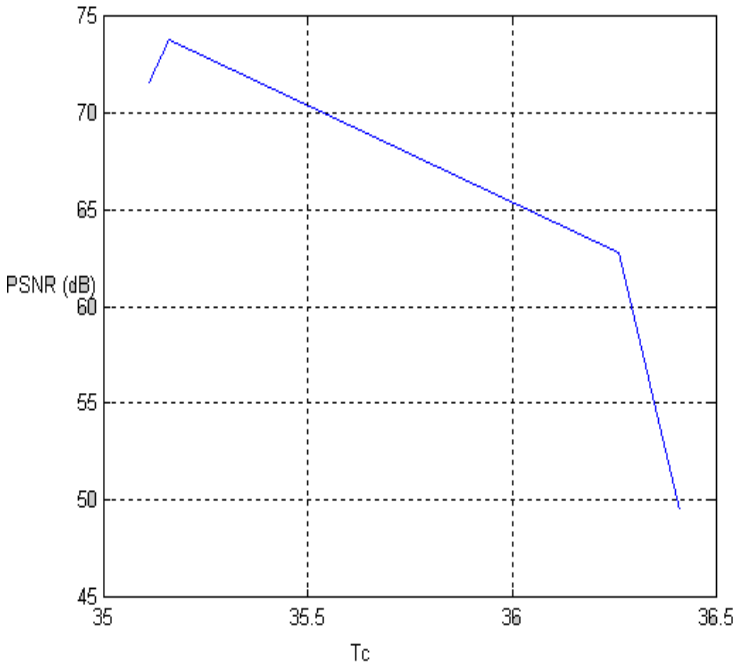
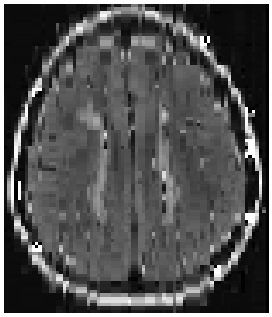


Figure IV-4: Variation du PSNR en fonction de Tc pour le 3eme cas

.



Tc=73,75

Tc=62,81

Figure IV-4-a: Images restituées pour le 3eme cas

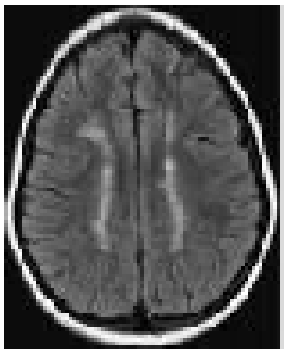


Image test

-Quatrième cas

Le tableau suivant indique la variation du taux de compression Tc et de PSNR de l'image de la figure 4-1en fonction de la taille des blocs m, en fixant le seuil T à 0,4, la taille du dictionnaire Tdico à 256et NQ=5.

m×m	PSNR (dB)	Tc (%)	Temps [H :Min:s :ms]
2×2	21,13	67,65	00 :00 :13 :11
4×4	21,65	65,93	00 :00 :12 :82
6×6	22,29	58,12	00 :00 :12 :80
8×8	22,44	39,53	00 :00 :12 :61

Tableau IV-4 : Résultats obtenus pour le quatrième cas

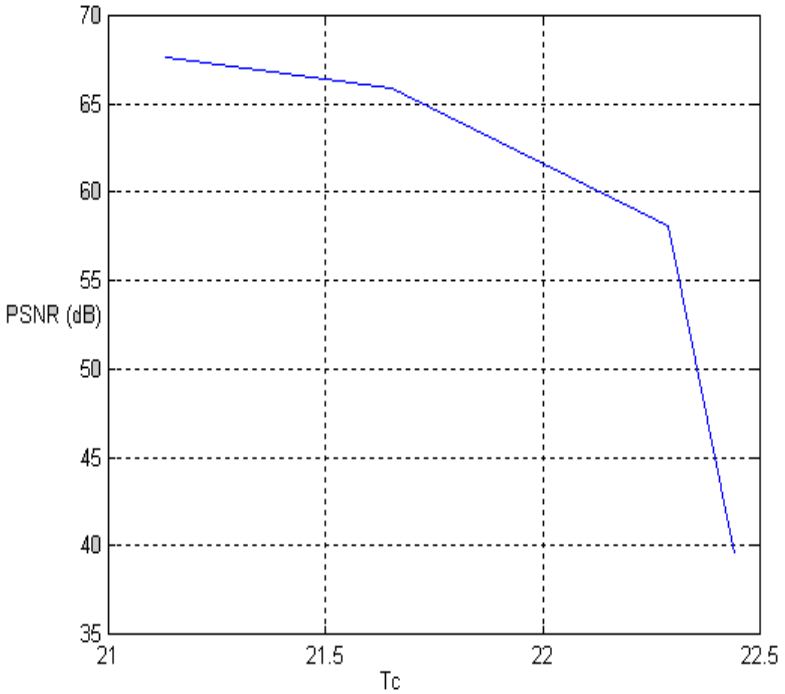
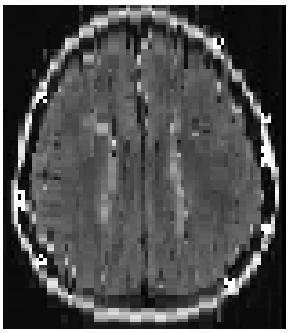
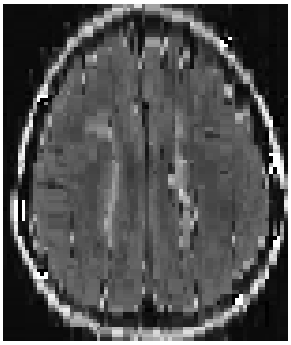


Figure IV-5: Variation du PSNR en fonction de Tc pour le 4eme cas



Tc=65,93



Tc=67,65

Figure IV-5-a: Images restituées pour le 4eme cas

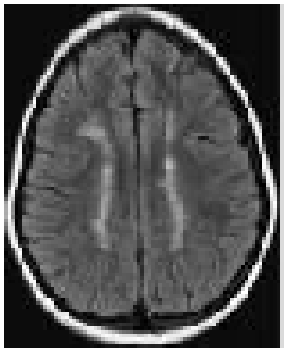


Image test

-Cinquième cas

Le tableau suivant indique la variation du taux de compression Tc et de PSNR de l'image de la figure 4-1en fonction de la taille des blocs m, en fixant le seuil T à 0,3, la taille du dictionnaire Tdico à 1024et NQ=1.

m×m	PSNR (dB)	Tc (%)	Temps[H :Min:s :ms]
2×2	28,50	43,91	00 :00 :14 :18
4×4	28,83	53,29	00 :00 :14 :12
6×6	29,22	37,83	00 :00 :13 :95
8×8	29,40	40.20	00 :00 :13 :89

Tableau IV-5 : Résultats obtenus pour le cinquième cas

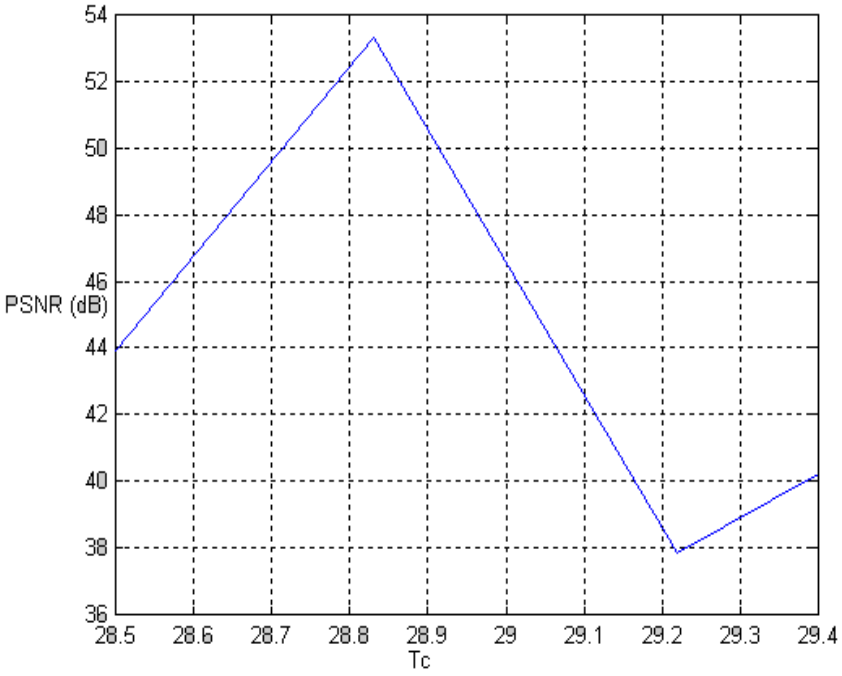
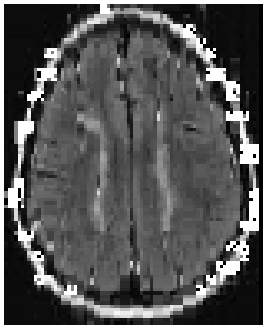
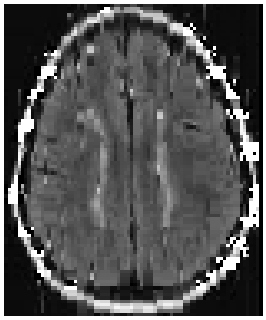


Figure IV-6: Variation du PSNR en fonction de Tc pour le 5eme cas



Tc=53,29



Tc=43,91

Figure IV-6-a: Images restituées pour le 5eme cas

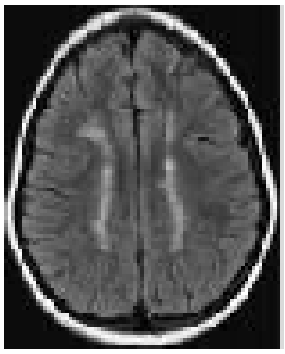


Image test

-Sixième cas

Le tableau suivant indique la variation du taux de compression Tc et de PSNR de l'image de la figure 4-1en fonction de la taille des blocs m, en fixant le seuil T à 0,3, la taille du dictionnaire Tdico à 1024et NQ=5.

m×m	PSNR (dB)	Tc (%)	Temps[H :Min:s :ms]
2×2	14,53	37,60	00 :00 :13 :91
4×4	14,85	43,83	00 :00 :13 :78
6×6	15,24	30,06	00 :00 :13 :68
8×8	15,42	31,27	00 :00 :13 :56

Tableau IV-6 : Résultats obtenus pour le sixième cas

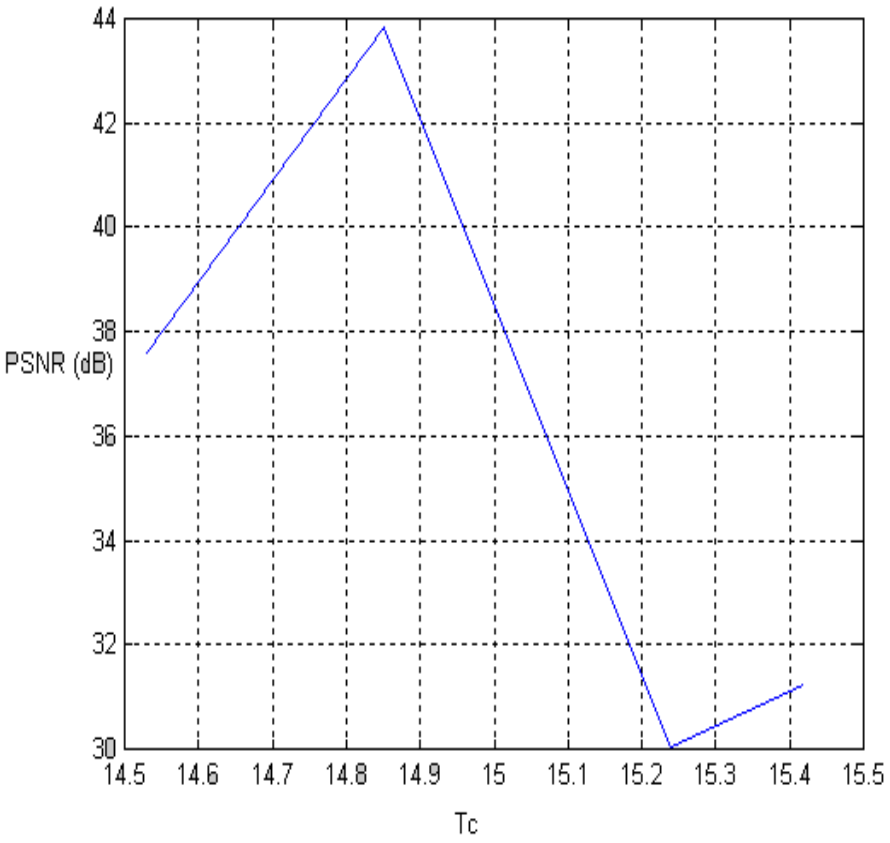


Figure IV-7: Variation du PSNR en fonction de Tc pour le 6eme cas

Tc=37,60

Tc=43,83

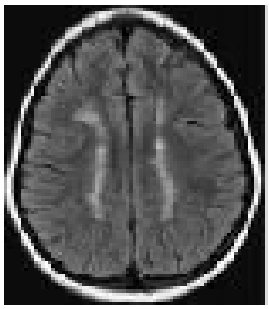


Image test

IV-4-Discussion des résultats

A partir des tableaux et des graphes que nous avons présenté nous remarquons que :

-Le PSNR est proportionnel à la taille des blocs m×m et inversement proportionnel à la taille du dictionnaire Tdico et à NQ .

-Le taux de compression Tc est inversement proportionnel a la taille des blocs m×m, à la taille du dictionnaire Tdico ainsi qu'à NQ.

-Le temps de calcul est d'autant plus grand que la taille du dictionnaire est grande,et il est d'autant plus petit que la taille des blocs m×m est petite.

-Les meilleures résultats du point de vue valeur PSNR, sont donnés pour une taille de dictionnaire Tdico=100, une taille des blocs m×m=8×8 et un pas de quantification NQ=1.

-Les meilleurs résultats du point de vue de l'aspect visuel sont donnés pour une taille de dictionnaire Tdico=256, une taille des blocs m×m=6×6 et un pas de quantification NQ=1.

Remarque :

Dans les tests que nous avons effectué, nous n'avons pas pu atteindre un seuil supérieur à 0.5 pour la classification ce qui est due au caractéristique de l'image car à partir d'un certain seuil qui est pour notre cas 0.5. L'image n'est plus divisible par deux zones.

IV-5-Conclusion

Dans notre étude, nous avons vue une méthode de compression qui est basée sur la classification d'une image en deux zones et l'application à ces dernières des deux méthodes de compression à savoir la TCD et la QV séparément.
En se referant aux différents résultats obtenus dans les divers cas étudiés, nous pouvons déduire que la qualité de l'image est liée directement au taux de compression et que ce dernier est inversement proportionnel à la taille des blocs m×m, à la taille du dictionnaire Tdico ainsi qu'à NQ.

Conclusion Générale

Annexes

I-MATLAB

MATLAB (MATrix LABoratory) est un logiciel de calcul scientifique dédié plus particulièrement aux applications numériques. A l'origine, il a été conçu pour manipuler des données matricielles ce qui en fait un outil majeur de l'analyse de données, du traitement d'image, de simulation numérique,…etc. Il possède son propre langage de programmation avec nombreuses fonctions fournies.

II-Commande de base

- help <commande> : Affiche une aide sur la commande.
- lookfor <mot> : Recherche les fonction contenant le mot dans leurs description.
- which <commande> : Affiche le chemin d'accès a la commande.
- who :Liste l'ensemble des variables de l'espace de travail .
- whos :Liste l'ensemble des variables ainsi que leurs taille .
- clear x :Supprime la variable x de l'espace de travail.
- clearall :Supprime l'ensemble des variables de l'espace de travail.
- clc :Effacer tout ce qui est écrit dans l'espace de travail.
- close all :Fermer toutes les fenêtres ouvertes.
- save : Sauvegarder l'ensemble des variables de l'espace de travail dans le fichier par défaut 'matlab.mat'.
- save :Sauvegarde l'ensemble des variables de l'espace de travail dans (fichier).
- load : Charge les variables contenues dans (matlab.mat) dans l'espace de travail.
- load : Charge les variables contenues dans (fichier).
- Path : Affiche la liste des chemins.
- Path (path,'d:/matlab') : pour ajouter un nouveau chemin d:/Matlab.
- Cd<chemin> : Afficher le chemin de répertoire de travail.

III-Toolbox Image

Image Processing Toolbox est une collection de fonction qui se base sur les capacités numériques de visualisation de MATLAB. C'est un boîtier qui contient une large gamme d'opérations sur le traitement d'image qui nous permet de :

- Analyser les images et faire des améliorations.
- Réaliser des opérations sur les images binaires.
- Réaliser des opérations sur les régions d'intérêt.

IV-Instruction utilisées

1-Charger une image

Pour lire une image sous MATLAB, on utilise l'instruction (imread).

Syntaxe : **imread ('l'emplacement de l'image')**

2-Visualiser une image

Pour la visualiser, on fait appel à l'instruction « imshow ».

Syntaxe : **imshow ('l'image')** ;

3-Histogramme de l'image

[Imhist] : donne l'histogramme d'une image en niveau de gris.

Syntaxe : **imhist (I)** ; **I** : étant l'image à binarisée.

4-Affichage des résultats sur des figures

Pour afficher l'histogramme de l'image, le résultat de la recherche …etc. on utilise la fonction « **plot** ».

Syntaxe : **plot (h)** ; **h** : étant l'histogramme de l'image.

Plot (h), hold on, plot (h1,'r')

Pour donner deux figures dans une même fenêtre.

Programme de la partie « classification »

```
 I=imread('chemin de l'image\nom de l'image.extension');
 I=rgb2gray(I);
I=double(I)/255;
[nblig1 nbcol1]=size(I);
figure(1), subplot(4,4,1), imshow(I);,title('image source');
blocs=im2col(I,[4 4],'distinct');
[nblig nbcol]=size(blocs);
x=blocs;
y=x;
 m=8;
    M=mean(blocs);
     v=var(blocs);
     v1=M;
 [n1,p1]=size(M);
 for i=1:p1
    if v1(i)==0
       v1(i)=0.000001;
    end
 end
    T=0.3;
 K=v./v1;
 % Homogene=[]
 % Heterogene=[]
 for i=1:nbcol
  if K(i)<= T
 x(:,i)=1 ;
  else
  x(:,i)=0 ;
    end
 end
```

```
n=0;
d=0;
for i=1:nbcol
   if x(1,i)==0
      n=n+1;
   else
      d=d+1;
   end;
end
nhm  =eye( nblig,n);
hm  = eye( nblig,d);
d=1;
f=1;
for i=1:nbcol
   if x(1,i)==0
        nhm(:,d)= y(:,i);
   d=d+1;
   else
     hm(:,f)= y(:,i);
     f=f+1;
   end
end
```

1-Imagerie médicale

L'imagerie médicale est le procédé par lequel un médecin peut examiner l'intérieur du corps d'un patient sans l'opérer. L'imagerie médicale peut être utilisée à des fins cliniques pour l'établissement d'un diagnostic ou pour le traitement de pathologies mais également dans le cadre de travaux de recherche scientifique étudiant la physiologie des êtres vivants.
L'importance que revêt l'imagerie médicale tient d'abord au fait qu'une image est un concentré d'informations bien plus efficaces qu'un texte ou qu'une explication verbale. Elle est certainement l'un des domaines de la médecine qui a le plus progressé ces vingt dernières années. Ces récentes découvertes permettent non seulement un meilleur diagnostic mais offrent aussi de nouveaux espoirs de traitement pour de nombreuses maladies. Cancer, épilepsie…

2-Principe de l'imagerie médicale

Le but de l'imagerie médicale est de créer une représentation visuelle intelligible d'une information à caractère médical. Cette problématique s'inscrit plus globalement dans le cadre de l'image scientifique et technique : l'objectif est en effet de pouvoir représenter sous un format relativement simple une grande quantité d'informations issues d'une multitude de mesures acquises selon un mode bien défini. L'image obtenue peut être traitée informatiquement pour obtenir par exemple :

-une reconstruction tridimensionnelle d'un organe ou d'un tissu ;

-un film montrant l'évolution ou les mouvements d'un organe au cours du temps ;

-une imagerie quantitative qui représente les valeurs mesurées pour certains paramètres biologiques dans un volume donné ;

-une représentation multimodale recalant plusieurs données au sein d'un même document (contour du cœur et mobilité des parois par exemple).

3-Les différentes techniques d'imagerie médicale

Suivant les techniques utilisées, les examens d'imagerie médicale permettent d'obtenir des informations sur l'anatomie des organes (leur taille, leur volume, leur localisation, la forme d'une éventuelle lésion, etc.) ou sur leur fonctionnement (leur physiologie, leur métabolisme, etc.). Dans le premier cas on parle d'imagerie structurelle et dans le second d'imagerie fonctionnelle.

Parmi les méthodes d'imagerie structurelles les plus couramment employées en médecine, on peut citer d'une part les méthodes tomographiques basées soit sur les rayons X (radiologie conventionnelle, tomodensitomètre ou CT-scan, angiographie, ...) soit sur la résonance magnétique (IRM) et, d'autre part, les méthodes échographiques (qui utilisent les ultra-sons).
Les méthodes d'imagerie fonctionnelles sont aussi très variées. Elles regroupent les techniques de médecine nucléaire (TEP, TEMP) basés sur l'émission de rayons gamma par des traceurs radioactifs qui, après injection, se concentrent dans les régions d'intense activité métabolique, les techniques électro-physiologiques qui mesurent les modifications de l'état électrochimique des tissus (en particulier en lien avec l'activité nerveuse) ou encore les mesures thermographiques.
Ces différents types de techniques sont souvent employés de façon complémentaire parfois même au sein d'un même système d'imagerie qui permet alors des acquisitions multimodales, simultanées ou non.

4-La neuro-imagerie : des images comme modèle

L'ambition des neurosciences est de faire correspondre à des activités complexes (souvent définies par l'approche psychologique en terme de résolution de problème : compréhension et production du langage, reconnaissance de formes, de visages, identification du sens d'un objet, formulation d'une intention…) des corrélats d'activité physiologique. Une fonction cognitive peut être définie comme **une séquence temporelle d'activités neuronales, électriques et neurochimiques, distribuées en réseau, et engendrant des variations locales à la fois du champ électro-magnétique, du métabolisme énergétique, et du débit sanguin cérébral (DSC)**.

Ce sont ces **marqueurs physiologiques** qui sont détectés par les techniques d'imagerie cérébrale : essentiellement la tomographie par émission de positons (TEP), l'imagerie par résonance magnétique fonctionnelle (IRMf) et la magnéto-électro-encéphalographie (MEEG).
La neuro-imagerie fonctionnelle donne ainsi accès au fonctionnement des réseaux entre différentes régions cérébrales et permet donc de proposer des modèles de fonctionnement du cerveau à un niveau le plus intégré possible. Les techniques d'imagerie cérébrale fonctionnelle postulent donc une ségrégation fonctionnelle du cortex cérébral en différentes aires. Dans cette hypothèse, l'exécution d'une fonction cérébrale correspond au recrutement spécifique d'un sous-ensemble de ces aires.

5-L'imagerie par résonance magnétique (IRM), généralités

L'IRM permet d'obtenir des images numériques en trois dimensions d'une précision inférieure au millimètre, du cortex, de la substance blanche, du liquide céphalo-rachidien et des noyaux gris centraux.
Initialement développé pour l'aide au diagnostic en milieu médical, ce type d'acquisition permet d'effectuer une analyse neuroanatomique individuelle de très haute précision.
L'IRM anatomique est utilisée depuis une vingtaine d'années dans le domaine médical, pour la précision des images anatomiques qu'elle fournie et donc des diagnostics qu'elle permet. Malgré leur coût élevé, quelque 22 000 appareils spécialisés fonctionnent actuellement dans le monde. Ils permettent chaque année de réaliser plus de 60 millions d'examens.

Le sigle IRM regroupe en fait un ensemble de techniques d'imagerie : outre des images anatomiques et fonctionnelles, l'IRM permet d'obtenir des images de l'organisation spatiale des vaisseaux cérébraux (angiographie IRM), de l'orientation des principaux faisceaux de fibres blanches (IRM de diffusion) , de la distribution de sang (IRM de perfusion) ainsi que de la concentration locale de certains métabolites énergétiques et peut-être, bientôt, de certains neurotransmetteurs (spectroscopie IRM).
C'est au début des années 70, grâce aux travaux du chimiste américain Paul Lauterbur et du physicien britannique Peter Mansfield – lauréats du Prix Nobel de médecine 2003 – ainsi que Raymond Damadian, que l'imagerie par résonance magnétique trouve un débouché médical.

Références Bibliographiques

Bibliographie

[1] : Yann Gaudeau ;
« Contributions en compression d'images médicales 3D et images naturelles 2D » ; Thèse de doctorat ; Université Henri Poincaré de Nancy 1 ; France ; 2006

[2] : M. Babel-Fouquet ;
« Compression d'images avec et sans perte par la méthode LAR » ; Thèse de doctorat ; INSA de Rennes ; France ; 2005

[3] : A. CZIHO ;
« Quantification vectorielle et compression d'images : application a l'imagerie médicale » ; Thèse de doctorat ; Université de Rennes 1 ; France ; 1999

[4]: « The transform and DATA compression handbook » ;Edited by Paularikas and R.C. Dorf; Uiversity of Texas at Arlington; 2001

[5] : Jean-Paul Guillois ;
« Techniques de compression des images » ; Edition Hermes ; 1996

[6]:S.Lounis, C.Djab ;
« Compression d'une séquence d'images par ondelettes » ; Thèse d'ingénieur d'état en électronique ; UMMTO ; 1999.

[7]:N. HAMRENE, D. IDIR, L. HAMOUDI ;
« Codage d'images en sous bande par fractales : appliqué aux images médicales » ; Thèse d'ingénieur d'état en électronique ; UMMTO ; 2005.

[8]:L. GUERDOU, F. OUHIB ;
« Codage d'images par régions d'intérêt » ; Thèse d'ingénieur d'état en électronique ;
UMMTO ; 2002.

[9]:Z. IDJERI, DJ. ISSOUN ;
« Compression d'images par paquets d'odelettes » ; Thèse d'ingénieur d'état en électronique ; UMMTO ; 2003.

[10]: AKROUR ;
« Compression d'images par fractales dans le domaine DCT » ; Thèse d'ingénieur d'état en électronique ; UMMTO ; 2003

[11] : Gilles BRUEL ;
« Introduction au traitement d'images \ simulation sous MATLAB» ; Edition Ermes Science ; 2001

[12] : P. Beaurepaire ;
« Codages d'images appliquée aux angiographies cardiaques » ; Thèse de doctorat ; Ecole Doctorale des sciences pour l'ingénieur de Lyon ; France ; 1997

[13] : htt://www.Kad chakib@infrance.com ;
« IFS et transformation fractale »

[14] : [1] : htt://www.médiathèque.image.fr ;Etude bibliographique des méthodes de compression d'images.