

UNIVERSITE DE SAAD DAHLEB DE BLIDA

Faculté des Sciences de l'Ingénieur
Département d'Electronique

MEMOIRE DE MAGISTER

Spécialité : Communication

COMPRESSION DES IMAGES FIXES PAR LA METHODE AMADCT

Par

M^{elle} BESSEKRI Anissa

Devant le jury composé de

M. Bensebti	Maître de Conférence, U. de Blida	Président
A. Guessoum	Professeur, U. de Blida	Examineur
Y. Smara	Professeur, U.S.T.H.B, Alger	Examineur
Z. Reguieg	Chargée de Cours, U. de Blida	Examinatrice
Z. Brahimi	Chargée de Recherche, CDTA	Rapportrice
K. Ait Saâdi	Chargée de Recherche, CDTA	Co- Rapportrice

Blida, Novembre 2005

RESUME

Dans ce travail, nous nous intéressons à la réduction du volume de données associées aux images fixes, sans toutefois perdre les informations utiles. Cette procédure est nommée compression. La compression représente une solution pour les problèmes majeurs rencontrés dans les applications de domaine de traitement d'images, généralement liés à la capacité de stockage et le temps de transmission des données. La compression peut être effectuée dans le domaine spatial ou dans un domaine transformé d'où l'existence de plusieurs techniques de compression avec ou sans perte d'informations lors de la reconstruction.

Les efforts des chercheurs du domaine ont permis le développement d'un grand nombre de méthodes de compression adaptatives basées sur les statistiques de données de l'image. Ces méthodes sont très efficaces pour le codage.

Dans ce mémoire nous avons développé un codeur adaptatif qui permet la compression des images. La compression se fait dans le domaine de la Transformée en Cosinus Discrète, ensuite nous effectuons une classification suivi d'une sélection des blocs de l'image. Par la suite la quantification des coefficients des blocs sera utilisée dans l'algorithme de codage de Huffman ainsi on obtient une image compressée. Les résultats obtenus montre l'efficacité de la méthode de compression étudiée.

ملخص

في هذا العمل، نهتم بتقليص حجم المعطيات الخاصة بالصور العددية الثابتة وهذا دون تضييع المعلومات المفيدة لهذه الصور. هذه العملية تسمى تخفيض حجم الصور. تخفيض حجم الصور يمثل حلاً للمشاكل الرئيسية المصادفة في تطبيقات ميدان دراسة الصور و المتعلقة أساساً بالقدرة على تخزين و زمن إرسال هذه المعطيات. عملية تخفيض حجم الصور يمكن القيام بها في المستوى الفضائي أو في المستوى المحول مما أدى إلى وجود عدة تقنيات لتبسيط الصور مع أو دون تضييع المعلومات و هذا عند استعادت الصور.

مجهودات الباحثين في الميدان سمحت بتطوير عدد كبير من طرق التخفيض المكيفة المعتمدة على إحصائيات معلومات الصورة. هذه الطرق هي جد فعالة من أجل المرقم.

في هذه المذكرة قمنا بتصميم مرقم متكيف يسمح تخفيض حجم الصور. عملية تخفيض الصور تعمل في مستوى المحول باستعمال نظرية التحويل العددي الجيبى دو البعدين (DCT)، بعدها نقوم بعملية ترتيب و اختيار الجمل، فيما بعد كل عناصر الجملة تعالج بواسطة مكمم. ترقم المعاملات المأخوذة فيما بعد بواسطة مرقم هوفمان. النتائج المحصل عليها تدل على فعالية طريقة تخفيض حجم المعطيات المدروسة

ABSTRACT

In this work, we interest ourselves with the reduction of the volume of data however associated with the fixed images, without to loss useful information. This procedure is named compression. Compression represents a solution to the major problems encountered in now application of image processing, generally related to stockage and the time capacity of transmission of the data. Compression can be carried out in the field space or a field technic of compression with or without loss of information at the time of rebuilding.

The efforts of the reseachers of field allowed the development of great number of adaptive methods of compression based on the statistics of data of the image. These methods are very effective for coding.

In this memory we developed an adaptive coder who allows the compression of the images. Compression is done in the field of the discret transform as a cosine, then we carry out classification followed of selection of the image areas. There after the quantification of the coefficients of the blocks will be used in the algorithm of coding of huffman thus one obtains an image compressed. The results obtained shows the effectiveness of methode of studied compression.

TABLE DES MATIERES

RESUME	
REMERCIEMENTS	
TABLE DES MATIERES	
LISTE DES ILLUSTRATIONS, GRAPHIQUES ET TABLEAUX	
INTRODUCTION	8
1. GENERALITES SUR LA COMPRESSION D'IMAGES	11
1.1 Introduction	11
1.2 Processus de codage / décodage	11
1.3 Domaine d'application de la compression d'images	18
1.4 Conclusion	18
2. TECHNIQUES DE COMPRESSION D'IMAGES FIXES	20
2.1 Introduction	20
2.2 Principe de la compression	21
2.3 Méthodes de compression d'images fixes	22
2.4 Techniques conservatives	23
2.5 Techniques non conservatives	27
2.6 Conclusion	43
3. METHODOLOGIE ET RESULTATS	44
3.1. Introduction	44
3.2. Algorithme de compression adaptative par la DCT-2D	45
3.3. Algorithme modifié de compression adaptative en DCT-2D	47
3.4. Algorithme modifié et amélioré de compression adaptative en DCT-2D	49
3.5. Décodage	55
3.6. Tests et résultats	59
3.7. Conclusion	70
CONCLUSION	71
RECOMMANDATIONS	73
REFERENCES	82

LISTE DES ILLUSTRATIONS, GRAPHIQUES ET TABLEAUX

Figure 1.1	Processus de codage / décodage	11
Figure 2.1	Synoptique d'un système de compression d'image	21
Figure 2.2	Synoptique d'un codec prédictif	28
Figure 2.3	Schéma de codage basé DCT	31
Figure 2.4	Diagramme de décomposition d'un signal 2D par ondelettes	32
Figure 2.5	Décomposition d'une image sur 2 niveaux de résolution	32
Figure 2.6	Décomposition en quadtree	33
Figure 2.7	Principe du quantificateur vectoriel	36
Figure 2.8	Exemple de quantification scalaire uniforme pour $L=5$	37
Figure 3.1	Transformation en DCT d'un bloc $n \times n$ pixels	45
Figure 3.2	Distribution de la probabilité de l'énergie AC dans un bloc	46
Figure 3.3	Schéma bloc de la ADCT	46
Figure 3.4	Schéma bloc de la MADCT	48
Figure 3.5	Schéma bloc de la MADCT améliorée	49
Figure 3.6	Table des pas de quantification	54
Figure 3.7	Balayage en Zig-Zag	55
Figure 3.8	Schéma de décodage	56
Figure 3.9	Schéma de codage	58
Figure 3.10	Résultat de l'application de l'approche MEM pour l'image Barbara avec un débit 0.42 bpp (PSNR =15.27 dB)	59
Figure 3.11	Résultat de l'application de l'approche MVC pour l'image Barbara avec un débit 0.80 bpp (PSNR =16.51 dB)	60

Figure 3.12	Résultat de l'application de l'approche MVC pour l'image côte d'azur avec un débit 0.83 bpp (PSNR =18.01 dB)	60
Figure 3.13	Résultats de l'application des deux approches pour l'image Volcan _ Irazu avec l'approche MOC-CPB avec un débit 0.82 bpp (PSNR =26.83 dB) et avec l'approche CPB avec un débit 0.82 bpp (PSNR =26.70 dB)	61
Figure 3.14	Résultats de l'application des deux approches pour l'image côte _ d'azur avec l'approche MOC-CPB avec un débit 0.81 bpp (PSNR =27.77 dB) et avec l'approche CPB avec un débit 0.81 bpp (PSNR =27.49 dB)	61
Figure 3.15	Résultats de l'application des deux approches pour l'image Nice avec l'approche MOC-CPB avec un débit 0.86 bpp (PSNR =25.97 dB) et avec l'approche CPB avec un débit 0.86 bpp (PSNR =25.89 dB)	62
Figure 3.16	Résultats de l'application des deux approches pour l'image Météo avec l'approche MOC-CPB avec un débit 0.80 bpp (PSNR = 30.24 dB) et avec l'approche CPB avec un débit 0.80 bpp (PSNR =30.13 dB)	62
Figure 3.17	Résultats de l'application des deux approches pour l'image Neurologique avec l'approche MOC-CPB avec un débit 0.76 bpp (PSNR =29.99 dB) et avec l'approche CPB avec un débit 0.76 bpp (PSNR =29.99 dB)	64
Figure 3.18	Résultats de l'application des deux approches pour l'image Radiologique avec l'approche MOC-CPB avec un débit 0.77 bpp (PSNR =37.57 dB) et avec l'approche CPB avec un débit 0.77 bpp (PSNR =37.50 dB)	64
Figure 3.19	Résultats de l'application des deux approches pour l'image Barbara avec l'approche MOC-CPB avec un débit 0.83 bpp (PSNR =31.30 dB) et avec l'approche CPB avec un débit 0.83 bpp (PSNR =31.06 dB)	66
Figure 3.20	Résultats de l'application des deux approches pour l'image Teffani avec l'approche MOC-CPB avec un débit 0.79 bpp (PSNR = 32.43 dB) et avec l'approche CPB avec un débit 0.79 bpp (PSNR =32.39 dB)	66
Figure 3.21	Résultats de l'application des deux approches pour l'image Bateau avec l'approche MOC-CPB avec un débit 0.80 bpp (PSNR =30.98 dB) et avec l'approche CPB avec un débit 0.80 bpp (PSNR =30.75 dB)	67
Figure 3.22	Résultats de l'application des deux approches pour l'image Volcan _ Irazu reconstruite avec classification avec un débit 0.82 bpp (PSNR=26.83 dB) et sans classification avec un débit 0.84 bpp (PSNR =27.15 dB)	68
Figure 3.23	Résultats de l'application des deux approches pour l'image Neurologique reconstruite avec classification avec un débit 0.76 bpp (PSNR=29.99 dB) et sans classification avec un débit 0.81 bpp (PSNR =32.06 dB)	69
Figure 3.24	Résultats de l'application des deux approches pour l'image Barabra reconstruite avec classification avec un débit 0.83 bpp (PSNR=31.30 dB) et sans classification avec un débit 0.84 bpp (PSNR =31.54 dB)	69

INTRODUCTION

L'image fait partie des moyens de communication que l'homme a toujours privilégié pour la richesse de son contenu. Son utilisation intensive dans les différents domaines importants comme les télécommunications, la médecine, la transmission de programmes de télévision, etc..., a rendu nécessaire le développement de diverses techniques de traitement, de transmission et de stockage.

Il y a quelques dizaines d'années que sont apparus les premiers systèmes utilisant l'image numérique comme support d'information. Depuis, ils n'ont cessé de se développer de plus en plus rapidement en raison des avantages offerts par les techniques de traitement d'images numériques. En effet, les images numériques ont un avantage indiscutable sur les images analogiques [1]. La numérisation des images facilite leur traitement et fiabilise leur transmission à travers des réseaux informatiques. Cependant, les images numérisées occupent un volume d'informations considérable, ce qui nécessite des dispositifs de stockage ainsi que, des bandes passantes de transmission suffisamment importantes. A titre d'exemple, une image fixe de dimension 512x512, et codée sur 8 bits par pixel, requiert un volume de données de 2. 097. 152 bits. En toute évidence, au travers de ces données, une banque d'archives devrait nécessiter un espace de stockage très important. Ceci est notamment le cas pour un service médical qui doit conserver des données images pour les patients. Il en est de même pour la conservation des images satellites qui seront utilisées pour établir des cartes géographiques ou météorologiques.

En dehors du problème de stockage, des bandes passantes très importantes devrait être envisagés d'un autre côté pour la transmission de ces données.

Pour pallier, notamment aux problèmes de la capacité de stockage et du temps de transmission, engendrés par l'énorme volume requis pour représenter les images numériques, le recours à des méthodes de compression est une nécessité accrue. Elles consistent à coder les images sous une forme dense représentant sensiblement le même message visuel.

L'efficacité de ces techniques est mesurée par leur capacité de compression avec ou sans distorsion et par la complexité de l'implémentation des algorithmes qui en découlent.

Les techniques de compression peuvent se classer en deux groupes en fonction de la qualité de l'image reconstruite. Le premier groupe concerne les techniques de compression sans pertes d'information, dites aussi réversibles ou à reconstruction exacte. Le second groupe a pour objectif de compresser l'image tout en acceptant une légère perte d'information. Pour cela, les algorithmes des techniques dites irréversibles, se basent sur la redondance psycho-visuelle inhérente à l'image afin de réaliser la compression. Ces méthodes tirent parti des imperfections du récepteur (œil) en privilégiant les classes d'informations auxquelles il est particulièrement sensible.

Des algorithmes adaptatifs de compression dans le plan transformé cosinus ont été développés par [2][3][4]. Deux applications sont envisagées dans le cadre de notre travail, un algorithme correspondant à la classe d'algorithmes adaptatifs et un algorithme de compression non adaptatif par DCT.

Le mémoire est organisé comme suit :

Le chapitre 1 : Dans lequel, une description des principaux composants d'un système de compression d'images est proposée, avec un aperçu sur les domaines d'application de la compression.

Le chapitre 2 : Il constitue un rappel sur les caractéristiques de corrélation ou de redondance du signal d'image. Il introduit les différentes techniques de compression des images et leur principe respectif.

Le chapitre 3 : Est consacré à la compression des images basée sur des algorithmes adaptatifs. Il couvre l'aspect conception et présentation des résultats de simulation. Les tests effectués vont porter sur trois types d'images (satellites, médicales et optiques). Les résultats obtenus vont faire l'objet d'une discussion pour évaluer les performances de l'approche adoptée.

Enfin, nous établirons une comparaison entre l'algorithme classique de compression sans classification et l'algorithme implémenté.

Nous terminons ce travail par une conclusion générale, qui récapitulera les objectifs de notre étude. Elle présentera aussi les éventuelles perspectives de ce travail.

CHAPITRE 1 GENERALITES SUR LA COMPRESSION D'IMAGES

1.1 Introduction

La compression d'images consiste à réduire le nombre de bits qui représente une image. Elle est connue sous le nom de codage. La décompression est l'opération inverse, qui permet de retrouver l'image d'origine. Elle est appelée décodage.

1.2 Processus de codage / décodage

La chaîne de codage et de décodage comporte les éléments suivants :

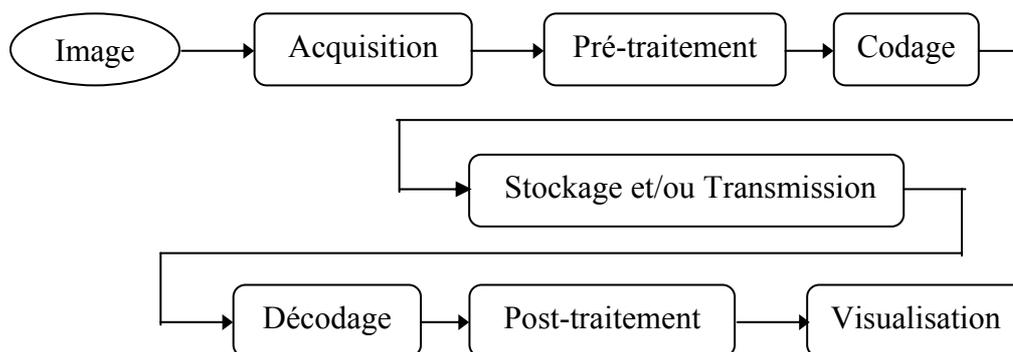


Figure 1.1 : Processus de codage / décodage

L'acquisition, la compression, la transmission, la décompression et la visualisation représentent les cinq maillons de base dans un système de compression, que ce soit dans le cadre d'une transmission ou d'un stockage des images.

1.2.1 Acquisition des données images

Le rôle d'un système d'acquisition est de transformer l'image du monde réel au monde numérique. Ceci est effectué grâce à une procédure de numérisation [5]. Ce système comporte

le plus généralement un dispositif d'acquisition et de numérisation, et un système de visualisation. Les fonctions assurées par chaque module du système sont les suivantes :

- La conversion du signal électrique (analogique) en un ensemble de données numériques : c'est la numérisation qui se fait par le biais d'un dispositif d'acquisition pouvant recevoir les signaux analogiques de la caméra, du magnétoscope, ou du scanner. Cette numérisation s'effectue par des opérations d'échantillonnage des coordonnées spatiales et de quantification des amplitudes de l'intensité lumineuse.
- La numérisation et la visualisation des données numériques : les informations binaires obtenues après conversion seront stockées dans un fichier.

1.2.2 Pré et Post- traitement

Le pré-traitement concerne les images acquises tandis que, le post traitement concerne les images codées. Dans les deux cas, les images ne peuvent pas échapper aux effets de dégradation dus essentiellement soient aux phénomènes physiques tels que - La diffraction du système optique, - Le flou dû au mouvement de l'image durant son acquisition. Soient aux erreurs survenues lors du processus de codage.

Le filtrage est l'une des méthodes développées pour pallier à ces dégradations. Il permet d'éliminer les parasites, tel que le bruit et d'améliorer la qualité des images. Les méthodes de filtrage sont des opérations locales qui consistent à remplacer la valeur de chaque pixel de l'image par une combinaison linéaire des valeurs de ses voisins. C'est la convolution. Elle consiste à faire balayer une fenêtre sur l'ensemble des pixels de l'image, afin de modifier la distribution fréquentielle des composantes du signal [6]. Nous distinguons le filtre passe-bas, le filtre passe-haut, le filtre passe-bande, et le filtre directionnel.

1.2.2.1 Filtre passe- bas

Cette opération consiste à éliminer le bruit contenu dans les hautes fréquences spatiales. Il est conçu pour atténuer les composantes hautes fréquences du signal d'image sans que l'information contenue dans les basses fréquences soit amputée : il adoucit considérablement les contours de l'image. Ce filtre de lissage consiste à remplacer la valeur

de chaque point par la moyenne effectuée sur un voisinage. Les filtres passe-bas fréquemment utilisés sont : Filtre moyen, Filtre moyen pondéré.

1.2.2.2 Filtre passe-haut

Il a les caractéristiques inverses du filtre passe-bas. Sa fonction de transfert atténue les fréquences basses sans toucher à l'information contenue dans les hautes fréquences. On utilise les filtres passe-haut, pour détecter et visualiser les limites entre les différentes zones de l'image.

1.2.2.3 Filtre passe-bande

Cette opération est une dérivée du filtre passe-bas. Elle consiste à éliminer la redondance d'information entre l'image d'origine et l'image obtenue par filtre passe-bas. Seule la différence entre l'image source et l'image traitée est conservée.

1.2.2.4 Filtre directionnel

Dans certains cas, on cherche à faire apparaître des détails de l'image dans une direction bien déterminée. Pour cela, on utilise des filtres qui opèrent suivant des directions (horizontales, verticales et diagonales).

1.2.3 Codage et Décodage

L'objectif de la compression d'images est de réduire le volume de données associé aux images afin d'optimiser les coûts de stockage et de communication. Les méthodes de compression d'images peuvent être classées selon deux critères liés à la distorsion apportée à l'image d'origine [7]. On distingue :

- Les méthodes sans pertes permettant de retrouver exactement les pixels de l'image numérique d'origine (pas de distorsion).
- Les méthodes avec pertes où l'image décompressée est une approximation de l'image d'origine. Ses distorsions sont généralement indécélables à l'œil nu.

Cependant, les méthodes sans pertes engendrent des taux de compression très faible et ne sont utilisés que dans des applications sensibles, telles que : les images médicales où l'image reconstruite doit montrer les structures fines de l'information fonctionnelle des organes des patients pour permettre un bon diagnostic. La compression des images se fait par voie logicielle ou bien, à l'aide d'une carte à processus Hardware insérée dans le micro-ordinateur. Pour la décompression, il y a aussi deux solution : soit c'est la carte qui le fait, soit c'est le logiciel.

1.2.4 Stockage et transmission

Un des aspects le plus important dans la conception d'un système de compression d'images est la possibilité de stockage de grand volume d'information. Les trois principaux médias de stockage utilisés sont le disque, les disquettes magnétiques et le disque optique.

1.2.5 Visualisation

La visualisation est l'une des opérations fondamentales dans un système de compression des images. Bon nombre de traitements ne sont jugés qu'en fonction de l'affichage et de la qualité visuelle du résultat. Les systèmes de compression utilisent pour leur besoin d'affichage des moniteurs couleurs ou monochromes T.V, clichés photographiques, impression sur papier.

1.2.5.1 Evaluation de la qualité de l'image décodée

La compression des images est un domaine qui connaît, depuis plusieurs années, une importance accrue, et qui conduit à une grande diversité d'approches et algorithmes. L'étude comparative des performances de ces techniques de codage a fait l'objet de nombreux travaux [8]. Généralement une méthode de compression est évaluée selon les critères suivant :

1.2.5.1.1 Degré de réduction des données

Le degré de réduction des données obtenu par une méthode de compression peut être évalué au moyen du quotient de compression défini par la formule [8] :

$$Q = \frac{\text{Taille des données initiales}}{\text{Taille des données compressées}} \quad (1.1)$$

On voit clairement que plus le quotient est élevée plus la méthode de compression employée est efficace.

1.2.5.1.2 Taux de compression

Etant donné que l'objectif d'une compression est de minimiser la quantité d'informations nécessaire à la représentation d'une image, on définit la quantité :

$$T_c (\%) = \frac{\text{Nombre de bits de l'image compressée}}{\text{Nombre de bits de l'image d'origine}} \times 100 \quad (1.2)$$

appelée taux de compression [8]. On utilise également la mesure

$$T_b = \frac{\text{Nombre de bits de l'image compressée}}{\text{Taille de l'image d'origine (pixels)}} \quad (1.3)$$

qui est donnée en bits par pixel (bpp).

1.2.5.1.3 Gain de compression

Le gain de compression est également exprimé en pourcentage; c'est le complément à 1 du taux de compression [8].

$$G(\%) = 100 - T_c (\%) \quad (1.4)$$

1.2.5.1.4 Mesure de fidélité

Le choix d'un système de codage résulte d'un compromis entre le taux de compression et la qualité des images reconstruites [8]. La recherche des modèles mathématiques permettant de simuler le comportement du système visuel, et de mesurer la qualité des images telles qu'il

les perçoit, reste le souci de nombreux laboratoires. Malheureusement, la complexité des traitements réalisés par le système visuel humain rend délicat l'obtention de modèles suffisamment précis, et le recours à une évaluation subjective de la qualité de l'image devient donc nécessaire.

1.2.5.1.4.1 Mesures subjectives

Des recherches ont permis de distinguer deux méthodes de mesure de la qualité, basées sur le calcul de l'opinion moyenne d'un groupe d'observateurs humains [9].

- Méthode de dégradation

Un groupe de personnes examine une séquence d'images, selon des conditions de visualisation définies par l'organisme de normalisation CCITT. Cette méthode présente aux observateurs des couples d'images, dans lesquelles la référence est annoncée et présentée en premier. L'image à juger est évaluée en terme de sévérité de dégradation selon une échelle donnée dans le tableau 1.1.

Catégorie	Dégradation	Qualité
5	Imperceptible	Excellente
4	Perceptible mais non gênante	Bonne
3	Légèrement gênante	Passable
2	Gênante	Médiocre
1	Très gênante	Mauvaise

Tableau 1.1 : Echelle de comparaison

- Méthode du double stimulus

Deux images A et B sont présentées. L'une est la référence, l'autre est à évaluer. L'observateur qui ignore laquelle est la référence évalue la qualité de A et B suivant une échelle bornée [9].

1.2.5.1.4.2 Mesures objectives

Les mesures de qualité objectives pour l'évaluation de l'image reconstruite, les plus utilisées sont [10] :

- L'erreur moyenne quadratique (MSE : Mean Square Error) :

La distorsion introduite lors du processus de codage/décodage de l'image, correspond à l'erreur quadratique moyenne entre l'image d'origine et l'image reconstruite [11]. Si l'on note I_0 l'image d'origine de taille $M \times N$, et I l'image de même taille obtenue après reconstruction, on peut définir :

$$\text{MSE} = \frac{1}{M \times N} \sum_{i=1}^{i=M} \sum_{j=1}^{j=N} (I_0(i,j) - I(i,j))^2 \quad (1.5)$$

Etant donnée que l'image reconstruite doit s'apprécier visuellement, on peut compléter l'information donnée par l'erreur quadratique moyenne en présentant « l'image différence » entre l'originale et la reconstruite.

Toutefois, la mesure normalisée de MSE donne de meilleurs résultats.

$$\text{NMSE} = \frac{\sum_{i=1}^{i=M} \sum_{j=1}^{j=N} (I_0(i,j) - I(i,j))^2}{\sum_{i=1}^{i=M} \sum_{j=1}^{j=N} (I_0(i,j))^2} \quad (1.6)$$

Une image décompressée est dite de bonne qualité si : $\text{NMSE} \leq 0.25$ [10].

- Le rapport signal maximal à bruit (PSNR : Peak Signal to Noise Ratio) :

L'évaluation de la qualité par cette mesure simple a montré une performance considérablement bonne [12]. Elle est donnée par :

$$PSNR = 10 \log_{10} \frac{V_{\max}^2}{V_{eqm}} \quad (dB) \quad (1.7)$$

où,

V_{\max} : l'intensité maximale (255 pour 8 bits/pixel)

V_{eqm} : L'MSE entre l'image d'origine et l'image reconstruite.

1.3 Domaine d'application de la compression d'images

L'image s'ouvre à des applications diversifiées allant des procédés d'archivage aux services de la visiophone, de la visioconférence et de la transmission de programmes de télévision. La compression d'images a véritablement percée dans tous les domaines de l'industrie : l'exploration fonctionnelle des modalités des images médicales, la photographie aérienne d'images satellites, télédétection.

La compression prend aujourd'hui une importance stratégique dans la vie des entreprises et des sociétés industrielles.

1.4 Conclusion

Durant les deux dernières décennies, les applications utilisant les images numériques se sont considérablement développées. L'image numérique, fixe ou animée, trouve en particulier ses applications dans :

la communication : courrier électronique contenant photos ou vidéos, visiophone ou visioconférence

les divertissements : télédiffusion, cinéma, documents ou jeux multimédias

la médecine : imagerie médicale (échographie, radiographie, scanner,...)

les interfaces homme-machine : commande naturelle de machines par le biais de la vision par ordinateur.

Cette émergence est due, d'une part au développement des technologies d'acquisition (appareil photo ou caméra numérique, scanner) et d'autre part à celui des technologies de traitement, stockage et transmission de données numériques.

Or, en la matière, les besoins augmentent plus rapidement que les moyens. En effet, les images fixes représentent des quantités gigantesques de données numériques qu'il faut réduire par des techniques de compression. Le chapitre suivant est consacré à l'étude détaillée des différentes techniques de compression d'images fixes.

CHAPITRE 2

TECHNIQUES DE COMPRESSION D'IMAGES FIXES

2.1 Introduction

La compression d'un signal numérique consiste à minimiser la quantité d'information nécessaire à sa représentation, en vue d'un stockage, d'une transmission. En traitement numérique d'images, les applications sont nombreuses et en rapide développement. Citons par exemple : image télévision, image satellite, image médicale, vidéo-conférence,...

Le problème crucial qui se pose pour l'utilisation de l'image est la grande quantité d'information ou de données nécessaires à leur représentation. En fait, le nombre de bits nécessaires pour représenter une information image peut être substantiellement réduit en raison de la redondance présente dans l'image. La redondance dans une image numérique peut être :

- Redondance spatiale (intra-image) :

Elle est basée sur la forte corrélation existant entre les pixels de l'image. Cette spécificité peut être mise à profit pour compacter les données. Cette corrélation ou dépendance est due à la faible variation de la luminosité d'un pixel à un autre [13]. La compression d'images fixes exploite ce type de redondance.

- Redondance temporelle (inter-image) :

Elle est liée à la forte corrélation existant entre les images successives d'une même séquence [14]. Ce type de redondance est exploité dans les images animées .

Tous ces types de redondance expriment l'intérêt de la minimisation de la quantité de données nécessaire pour la représentation de l'information image. C'est pour ces raisons que de multiples techniques de compression sont développées.

2.2 Principe de la compression

Le principe général de toute méthode de compression d'images est donné par la figure 2.1 :

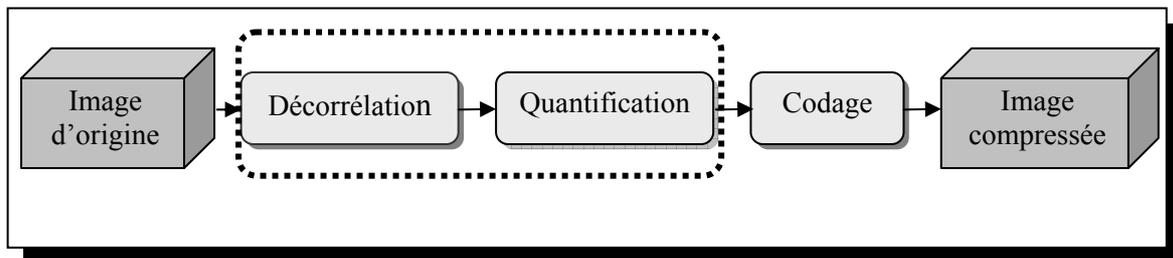


Figure 2.1 : Synoptique d'un système de compression d'image

2.2.1 Décorrélacion

La dépendance existante entre les pixels voisins (la luminosité varie très peu d'un pixel à un pixel voisin) se traduit par une très forte corrélation sur l'image. Il s'agit donc de tirer profit de cette corrélation, pour réduire le volume d'information en effectuant une opération de décorrélacion des pixels. Cette dernière consiste à transformer les pixels initiaux en un ensemble de coefficients plus ou moins décorrélés [10], à partir duquel la reconstruction de l'image est possible. Cette opération est réversible.

2.2.2 Quantification

La quantification des coefficients a pour but de réduire le nombre de bits nécessaire pour leur représentation. Elle représente une étape clé de la compression. C'est une application surjective (plusieurs points possèdent le même niveau quantifié) et c'est cet aspect qui le rend non réversible. La quantification peut être appliquée sur un seul élément (quantification scalaire), ou sur un groupe d'éléments (quantification vectorielle).

2.2.3 Codage

Le codage est la dernière étape de la compression, il consiste à coder les coefficients quantifiés, selon leurs fréquences d'apparition, en leurs attribuant des mots binaires. Le codage utilisé a une influence importante sur le gain de compression.

2.3 Méthodes de compression d'images fixes

Beaucoup de méthodes classiques ont été développées pour réduire l'information, tout en conservant l'information utile. Ces méthodes de compression peuvent être divisées en deux familles distinctes :

- La compression conservative (réversible).
- La compression non conservative (non réversible).

Les expérimentations menées montrent que généralement les méthodes qui atteignent des taux de compression très élevés sont les méthodes avec distorsion. Par contre, les méthodes sans distorsion engendrent des taux de compression faibles et ne sont utilisées que dans des applications sensibles qui ne tolèrent pas de perte d'informations telles que les images satellitaires et les images médicales.

2.3.1 La compression conservative (sans distorsion)

Dans cette classe de compression, toute sorte de perte d'informations est impardonnable, donc le processus de décompression doit générer une copie exacte des données lues avant la compression.

2.3.2 La compression non conservative (avec distorsion)

Les techniques non conservatives ont été largement utilisées dans les applications de compression des données, dont la qualité se limite aux perceptions humaines (images, vidéos, sons,..). Elles sont différentes des techniques conservatives : elles acceptent une légère perte d'informations, afin d'obtenir un taux de compression le meilleur possible.

2.4 Techniques conservatives

Les algorithmes de compression / décompression utilisés se doivent être non dégradants pour garantir l'intégrité des informations traitées. Parmi celles-ci, on trouve :

2.4.1 Méthodes statistiques (entropique)

En tenant compte des caractéristiques d'entropie des données à traiter, il est donc possible de définir des codages plus performants. Les travaux de Shannon [15] et de ses collaborateurs ont conduit au développement des algorithmes de codage statistique fondés sur la fréquence d'apparition d'une information.

Considérons une image dont chaque pixel est quantifié sur 8 bits. L'image possède 2^8 niveaux de gris différents. Dans la littérature, on parle de loi de codage par mots de longueur variable VLC (Variable Length Code).

Le codage par entropie est une technique de codage efficace. Son but est de coder les pixels en se basant sur des mots de codes de longueur variables L , égale à $-\log_2 p_i$ bits, de manière à ce que le nombre moyen de bits par pixel soit égal à l'entropie H (où p_i est la probabilité du pixel i). La limite de l'entropie H peut être approchée en codant des séquences de bits. Parmi les codes optimaux existants, le code de Huffman est le plus simple et le plus utilisé dans le codage des images. En effet, il satisfait à la loi du préfixe, ce que implique que la suite binaire peut être décodée sans ambiguïté.

2.4.1.1 Codage entropique de Huffman

Le codage de Huffman crée des codes à longueurs variables sur un nombre entier de bits [16]. L'algorithme considère chaque élément à coder comme étant une feuille d'un arbre qui reste à construire. L'idée est d'attribuer à chacun des ces éléments un code binaire dont la taille est d'autant plus réduite que le code est fréquent.

La procédure de codage construit un arbre dont les suites de 0 ou 1 constituent les mots de code. Elle s'effectue de la façon suivante [10] :

1. Ordonner la suite de probabilités $\{ P_i \}$ sans prendre en compte les probabilités nulles.
2. Additionner les deux probabilités les plus faibles, puis réordonner la nouvelle suite. Itérer le processus jusqu'à l'obtention d'une suite à deux éléments (en cas d'égalité de probabilités, l'ordre est arbitraire).
3. Construire l'arbre de codage en partant de la dernière suite et en affectant le code zéro à chaque branche gauche, et le code 1 à chaque branche droite.

2.4.1.2 Codage arithmétique

Il remplace un flot de symboles par un seul nombre, en virgule flottante compris entre 0 et 1. Le processus de codage restreint l'intervalle des nombres possibles (initialement $[0, 1]$) à chaque nouveau symbole [10]. Le nouvel intervalle est proportionnel à la probabilité prédéfinie attachée à ce symbole. Le nombre ainsi obtenu peut être décodé de façon unique.

Les procédures de codage arithmétique utilisent des tables décisionnelles de codage arithmétique qui peuvent également être déterminées par une spécification de table. A la réception, la restitution de l'image initiale est obtenue par l'opération de décodage, cette reconstruction doit être d'une grande fidélité.

2.4.2 Le codage différentiel

Le codage différentiel exploite la corrélation des valeurs de pixels voisins. Au lieu de coder les pixels, il code juste la différence entre le pixel et son précédent. Ces différences sont généralement représentées sur un nombre de bits plus petit que les valeurs des pixels.

2.4.3 Codage par répétition

Les méthodes Run Length consistent à coder les octets répétés sous forme d'un seul octet, suivi du nombre de répétitions. Cette méthode non dégradante est très simple, mais donne néanmoins de bons résultats pour certains types d'images (dessins au trait, images monochromes) [17].

2.4.3.1 Méthode par plage (RLC)

Le codage RLC (Run Length Coding) consiste en une description de l'image ligne par ligne ou colonne par colonne en codant des plages de pixels de même niveau de gris. Le codage d'une image par la méthode RLC opère de la manière suivante [7] :

- On émet un bit pour dire si l'on décrit l'image par ligne ou par colonne,
- Pour chaque séquence de pixels consécutifs de même niveau de gris, on émet le vecteur de deux dimensions $p_i = (m, n)$.

Où :

m : nombre de pixels voisins possédant le même niveau de gris.

n : niveaux de gris commun relatif à la séquence m .

Cette méthode est d'autant plus efficace que le nombre de niveaux de gris possible est faible. Elle s'applique avec une très grande efficacité aux images binaires.

2.4.3.2 Le codage de séquences nulles (RLE)

Le codage RLE (Run Length Encoding) est un cas particulier du codage RLC où on ne code que les valeurs nulles [7]. Pour cela un compteur est initialisé à zéro, pour chaque valeur nulle rencontrée dans la source, le compteur est incrémenté. Dès qu'une valeur non nulle est atteinte un code RLE sera émis.

2.4.4 Codage par dictionnaires

Le codage par dictionnaires est également fondé sur l'analyse des répétitions dans les données à traiter. Cependant, il ne s'agit plus ici de rechercher des occurrences des signifiants considérés comme élémentaires (généralement des octets), mais de "mots "(groupes d'octets) de longueur variable. Les mots répétés prennent place dans un dictionnaire, et chacun d'eux est remplacé, dans les données compressées, par sa seule adresse dans le dictionnaire.

Abraham Lempel et Jacob Ziv ont publié en 1977 et 1978 des algorithmes de compression d'usage général fondés sur la construction d'un dictionnaire dynamique indépendant de la nature des données codées. Par ailleurs, sa reproduction automatiquement par le décodeur est en fonction des données du fichier compressé [15].

2.4.4.1 Les méthodes LEMPEL-ZIV (LZ , LZW)

Les algorithmes de compression et décompression LZ ont des fonctionnements symétriques. Leur principe est fondé sur l'indexation de chaînes dans un dictionnaire qui, dans les deux cas est construit durant le traitement.

Le dictionnaire est défini comme un tableau de chaînes de tailles variables, repérées par leur adresse; la taille du tableau est également variable, limitée par le mode de codage des adresses. La programmation de cette méthode est assez complexe (gestion de pointeurs « glissants » sur des fenêtres, gestion de tableaux de longueur variable composés d'objets de longueur variable), et son programme peut conduire à des délais de traitement longs notamment en compression.

La compression LZW

Abraham Lempel et Jakob Ziv ont créé le compresseur LZ77 en 1977 (d'où son nom). Il était alors utilisé pour l'archivage (les formats ZIP, ARJ et LHA l'utilisent). En 1978, ils créent le compresseur LZ78 spécialisé dans la compression d'images (ou tout type de fichier

de type binaire). En 1984, Terry Welch le modifia pour l'utiliser dans des contrôleurs de disques durs, son initial vint donc se rajouter à l'abréviation LZ pour donner LZW [18].

LZW est un algorithme très rapide aussi bien en compression qu'en décompression, il substitue des motifs en construisant au fur et à mesure un dictionnaire. De plus, il se base sur des bits et non sur des octets, il ne dépend donc pas de la manière avec laquelle le processeur code les informations. C'est un des algorithmes les plus populaires, il est utilisé pour les formats TIFF et GIF.

Construction du dictionnaire

Le dictionnaire est initialisé avec les 256 valeurs de la table ASCII. Le fichier est découpé en chaîne d'octets (ainsi pour des images monochromes – codées sur 2 bits – cette compression est peu efficace), chacune de ces chaînes est comparée au dictionnaire et est ajoutée si jamais elle n'y est pas présente. Pour la décompression l'algorithme reconstruit le dictionnaire dans le sens inverse, il n'a donc pas besoin d'être stocké.

Compression

L'algorithme parcourt le flot d'information en le codant, si la chaîne est plus petite que le grand long mot du dictionnaire, elle est transmise.

2.5 Techniques non conservatives

On les appelle aussi les techniques non réversibles. Elles tirent partie d'une légère modification des données initiales pendant le cycle de compression/décompression, sans affecter gravement la qualité de l'image perçue par l'utilisateur. Parmi les divers techniques qui existent. Nous citons :

- ✓ Le codage prédictif.
- ✓ Le codage par transformée (DCT, Ondelettes, Fractales, ...etc.).
- ✓ Le codage par quantification vectorielle.
- ✓ Le codage pyramide.
- ✓ Le codage hybride.

2.5.1 Codage Prédicatif

Les méthodes prédictives sont considérées comme l'approche la plus ancienne. Elles exploitent la corrélation qui existe entre les pixels adjacents, qui sont liés les uns aux autres aussi bien sur le plan spatial que temporel. La prédiction peut être effectuée sur la base d'un pixel ou d'un bloc de pixel, où la même fonction de prédiction est appliquée à chaque pixel du bloc.

Un prédicteur peut être classé comme linéaire ou non linéaire, suivant la fonction de prédiction qui lui est associée.

Une autre classification qui dépend de la localisation des pixels précédemment utilisés peut être définie : il s'agit des prédicteurs intra-image et des prédicteurs inter-image [13][17].

Le prédicteur intra-image : utilise les pixels précédemment traités dans la même image, il est dit unidimensionnel quand il utilise les pixels précédents dans la même ligne et bidimensionnel quand il utilise les pixels de la ligne précédente.

Dans le but d'exploiter la corrélation temporelle et d'éliminer la redondance d'une image à l'autre, le prédicteur inter-image utilise une combinaison de pixels appartenant à l'image courante et à l'image précédente.

Le codage par prédiction est une méthode de compression avec perte, son principe est le suivant :

- Le prédicteur établit une estimation (prédiction) de la valeur de chaque pixel à partir des pixels déjà traités. La différence entre la valeur prédite et la valeur du pixel transmis est appelée erreur de prédiction.
- Cette dernière est quantifiée, codée par un codeur entropique ou arithmétique.
- En recevant le mot de code, le décodeur récupère l'erreur de prédiction, l'ajoute à la valeur calculée par son propre prédicteur et restitue la pixel correspondant.

Un synoptique du système de codage/décodage prédictif est décrit dans la figure 2. 2.

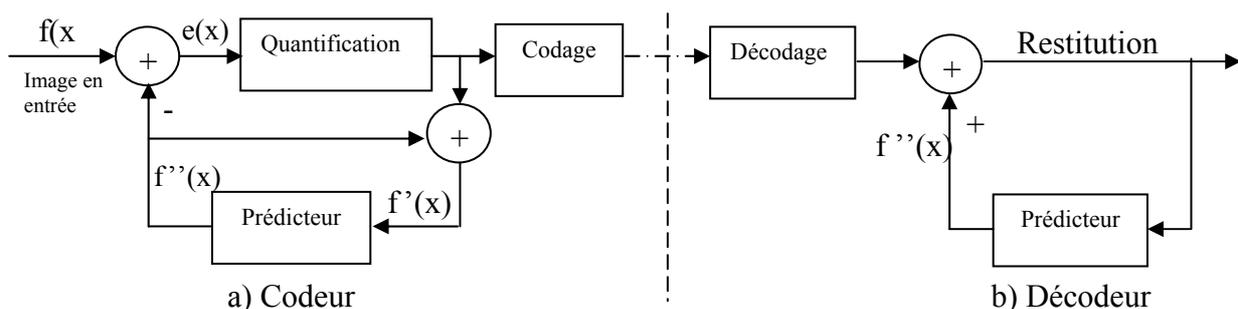


Figure 2.2 : Synoptique d'un codec prédictif

Où

$f(x)$: valeur originelle du pixel.

$f'(x)$: valeur reproduite de pixel (la même que celle du décodeur).

$f''(x)$: valeur prédite du pixel à partir des pixels déjà traités.

$e(x)$: erreur de prédiction.

Parmi les techniques prédictives, l'approche la plus utilisée est la Modulation par Impulsions Codées des Différences (MICD) [19], connue sous le nom DPCM (de l'anglais Differential Pulse Code Modulation).

2.5.2 Le codage par transformée

Son but est d'avoir une représentation de données dans un autre domaine, favorisant un codage plus efficace. Ce codage se base sur le fait que les propriétés de décorrélation entre les pixels de l'image qui se manifestent par des propriétés de concentration de l'information sur un nombre réduit de coefficients dans le domaine transformé [1].

Parmi les transformations linéaires qui existent, on peut citer [10] :

1. Transformation de Karhunen – loeve (TKL).
2. Transformation de Fourier discrète (TFD).
3. Transformation de Walsh – Hadamard (WAT) (TH).
4. Transformation de Haar (THA).
5. Transformation de Slant .
6. Transformation en Ondelettes (TO).
7. Transformation en Cosinus discrète (DCT).
8. Transformation en Sinus discrète (DST).
9. Transformation de Hartley (THR).
10. Transformation affines (Fractales).

2.5.2.1 Transformation en cosinus discrète (DCT)

La transformation en cosinus discrète (DCT) est la plus utilisée pour la compression des images [20]. C'est une transformation bidimensionnelle avec une fréquence qui augmente dans les directions horizontales et verticales. Sa formule mathématique est définie par :

$$G(u, v) = \frac{1}{\sqrt{2N}} \alpha(u) \alpha(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \cos\left(\frac{(2x+1)u\pi}{2N}\right) \cos\left(\frac{(2y+1)v\pi}{2N}\right) \quad (2.1)$$

avec $u, v = 0, 1, \dots, N-1$

$$\text{où } \alpha(i) = \begin{cases} 1/\sqrt{2} & \text{pour } i = 0 \\ 1 & \text{pour } i = 1, 2, \dots, N-1 \\ 0 & \text{ailleurs} \end{cases}$$

De la même façon, on définit la transformée inverse:

$$f(x, y) = \frac{1}{\sqrt{2N}} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} \alpha(u) \alpha(v) G(u, v) \cos\left(\frac{(2x+1)u\pi}{2N}\right) \cos\left(\frac{(2y+1)v\pi}{2N}\right) \quad (2.2)$$

avec $x, y = 0, 1, \dots, N-1$.

x et y étant les coordonnées dans le domaine spatial, u et v dans le domaine fréquentiel.

Son efficacité vient du fait que les coefficients transformés d'amplitude non négligeable se trouvent statistiquement concentrés dans une région restreinte aux basses fréquences spatiales, par conséquent les hautes fréquences seront éliminées de la phase de quantification et de codage.

A l'origine, c'étaient Huang et Schultheiss [21] qui ont proposé le principe de ce codage (Figure 2.3). L'idée est de découper l'image de dimension $N \times N$ en sous images ou blocs de taille réduite et à appliquer pour chaque bloc de l'image une transformation mathématique orthogonale linéaire du domaine spatial vers le domaine fréquentiel, indépendamment des autres blocs (transformée en un ensemble de coefficients plus ou

moins indépendants). Les coefficients obtenus de la phase de transformation sont quantifiés afin d'avoir un nombre de données considérablement réduit. Les informations quantifiées seront codées selon un codage de Huffman ou arithmétique. Pour retrouver l'intensité des pixels initiaux, on applique sur les coefficients transformés la transformation inverse.

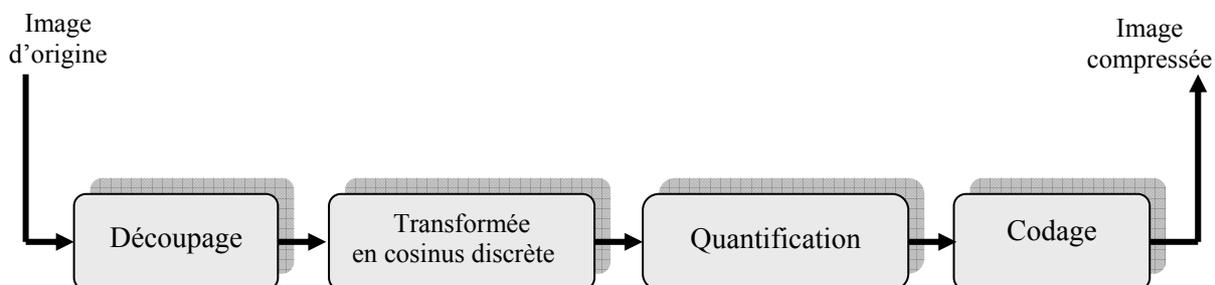


Figure 2.3 : Le schéma de codage basé DCT

2.5.2.2 Transformation en ondelettes (TO)

La transformée en ondelettes apparaît comme une technique d'avenir pour la compression d'images quel que soit le type d'applications [22].

La transformée en ondelettes est une transformée spatio–fréquentielle. Elle permet de réorganiser l'information contenue dans l'espace d'origine sur différentes plages spatiales et fréquentielles. Un des principaux avantages de la transformée en ondelettes est la décomposition d'une image en sous_images de différents niveaux de résolution, qui possèdent des caractéristiques statistiques intéressantes pour la compression (entropie, variance..) et peuvent être codées de manière indépendante.

La décomposition de l'image d'origine en sous–bandes est réalisée en se basant sur la théorie de l'analyse multirésolution introduite par Stéphane Mallat [23] et de l'analyse multirésolution biorthogonale [24] appliquée à la compression d'images : des filtres calculés à partir de bases d'ondelettes sont appliqués sur l'image. La décomposition sur un niveau s'effectue par application de filtres passe bas h et passe haut g sur l'image source. Cette opération est réalisée en appliquant de manière indépendante sur les lignes et les colonnes de l'image les filtres h et g (Figure 2.4). Après chaque filtrage, on procède à un sous_échantillonnage des données, en conservant un échantillon sur deux. L'image est donc divisée d'un niveau de résolution en quatre sous-bandes, l'une étant une approximation à la

résolution inférieure de l'image, les trois autres correspondent aux images des contours verticaux, horizontaux et diagonaux. Le passage aux résolutions suivantes s'effectue en appliquant le même type de filtrage sur chaque approximation de l'image (Figure 2.5). La compression proprement dite consiste à ignorer les informations les moins significatives.

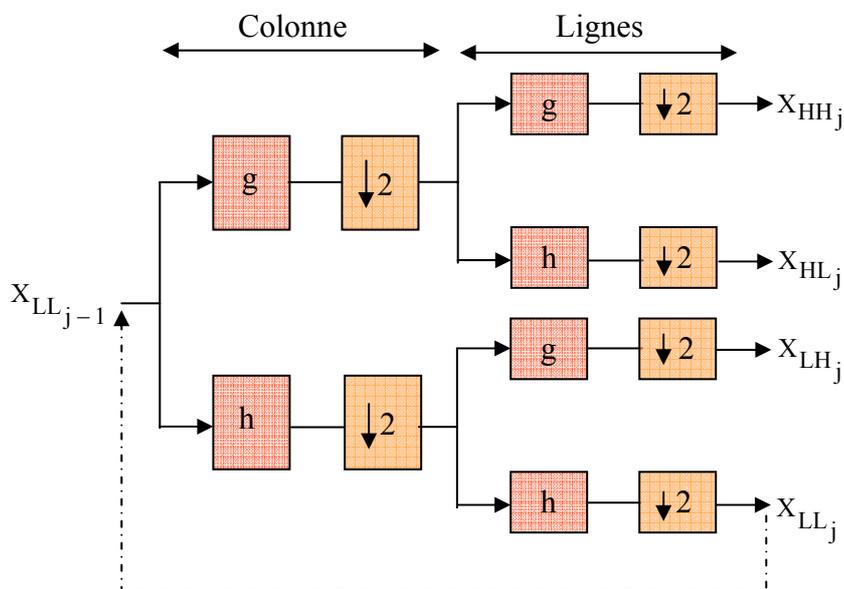


Figure 2.4 : Diagramme de décomposition d'un signal 2D par ondelettes

Les sous images obtenus après filtrage auront la taille réduite au quart.

HH_j : est la sous image isolant les hautes fréquences horizontales, verticales et diagonales.

HL_j : est la sous image isolant les hautes fréquences horizontales.

LH_j : est la sous image isolant les hautes fréquences et verticales.

LL_j : est la sous image représentant l'approximation de l'image originale à la résolution 1/2.

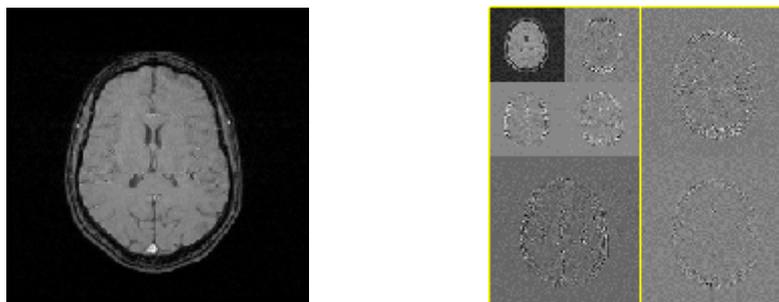


Figure 2.5 : Décomposition d'une image sur 2 niveaux de résolution

Le succès des schémas basés sur les ondelettes est essentiellement attribué aux stratégies innovatrices pour l'organisation et la représentation des coefficients d'ondelettes des images transformées.

Les principales méthodes de compression basées sur la transformée en ondelettes sont celles basées sur l'exploitation de la structure d'arbre des coefficients telles que EZW (Embedded Zerotree wavelet) de Shapiro [25], SPIHT (Set Partitioning In Hierarchical Trees) de Said et Al [26].

2.5.2.2.1 Algorithme EZW

Dans la méthode basée sur les ondelettes imbriquées en arbres (EZW) [25], les coefficients de l'ondelette sont regroupés dans les SOT (Arbres d'Orientation Spatiale) comme illustré sur la figure 2.6. L'amplitude de chaque coefficient d'ondelette dans un arbre, commençant avec la racine de l'arbre situé dans la bande de décomposition LL est ensuite comparée à certain un seuil T . Si l'amplitude de tous les coefficients d'ondelettes dans l'arbre sont plus petits que T , la structure entière de l'arbre (c'est à dire la racine et tous ses nœuds descendants) est représentée par le symbole ZTR, qui est le symbole de l'arbre zéro.

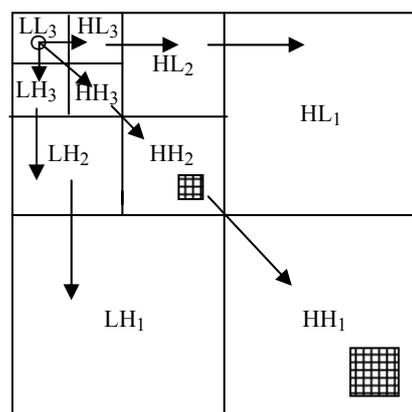


Figure 2.6 : Décomposition en quadtree

Si cependant, il existe dans l'arbre des coefficients d'ondelette importants (plus grand que T) alors la racine de l'arbre est représentée comme étant importante (représentée par le symbole

SIG quand son amplitude est plus grande que T), ou insignifiant (le symbole IZ, zéro isolé, quand l'amplitude est inférieure à T). Les nœuds descendants sont ensuite examinés chacun à son tour pour déterminer si chacun est la racine d'une possible structure sub_zéro ou non. Ce processus continue de telle manière que tous les nœuds dans tous les arbres sont examinés pour des structures sub_zéro possibles. Les coefficients d'ondelettes importants dans un arbre sont représentés par un ou deux symboles POS ou NEG, selon que leurs valeurs respectives soient positives ou négatives. Le processus de classification des coefficients comme étant ZTR, IZ, POS ou NEG est appelé la passe dominante. Ceci est ensuite suivi par une passe subordonnée dans laquelle les coefficients conséquents d'ondelettes dans l'image sont affinés en déterminant si leurs amplitudes se situent dans les limites des intervalles $[T, 3T/2[$ et $[3T/2, 2T[$. Les coefficients d'ondelettes dont les amplitudes se situent dans l'intervalle $[T, 3T/2[$ sont représentés par le symbole 0 (LOW), alors que ceux dont les amplitudes se situent dans l'intervalle $[3T/2, 2T[$ sont représentés par le symbole 1 (HIGH). A la suite de l'achèvement des deux passes dominante et subordonnée, la valeur du seuil T est réduite par un facteur de 2, et le processus est répété. Cette stratégie de codage consistant en une passe dominante et une subordonnée suivie d'une réduction de la valeur du seuil, est réitérée jusqu'à l'obtention du seuil voulue. Cette réduction de seuil agit essentiellement comme un quantifiable uniforme des coefficients. La séquence dans laquelle les coefficients sont examinés est prédéfinie par le codeur et connue par le décodeur. On y fait parfois référence comme un ordre de scanning d'arbre zéro. Par conséquent, la séquence de bits compressée se compose du seuil initial et des symboles d'arbres. Ces informations sont codées en binaire en utilisant un codeur entropique. Un décodeur EZW reconstruit l'image en actualisant progressivement les valeurs de chaque coefficient d'ondelettes dans un arbre au fur et à mesure qu'il reçoit les données. Les décisions du décodeur sont toujours synchronisées avec celles de l'encodeur.

2.5.2.2.2 Algorithme SPIHT

SPIHT est similaire à EZW, en ce sens qu'il exécute un rangement des coefficients en utilisant un ensemble de seuils décroissant en octave [26]. Dans ce cas le seuil initial correspond à la plus grande puissance de deux qui est plus petite que l'amplitude du plus grand coefficient. En outre, SPIHT utilise un SOT différent, qui est basé sur la supposition que : la dimension de la bande LL générée par l'ondelette de décomposition est un entier multiple de deux. Cela permet un groupement de 2 par 2 des coefficients adjacents dans la

bande LL. Dans chaque groupe, un coefficient est choisi pour ne pas avoir de descendants dans les autres sous bandes, alors que les coefficients restants ont des descendants.

SPIHT regroupe les coefficients en trois listes, la liste des ensembles insignifiants (LIS), la liste des pixels insignifiants (LIP), et la liste des pixels importants (LSP). Ces listes sont initialisées respectivement à un ensemble de descendants de sous arbres des nœuds du plus haut niveau, les nœuds du plus haut niveau est une liste vide. Un processus de deux phases qui consiste en une passe de tri et une passe d'affinement est ultérieurement initiée. Pendant la passe de tri (rangement), l'algorithme parcourt le LIP testant l'amplitude de ses éléments relativement au seuil courant et représentant leur importance par un 0 ou 1. Chaque fois qu'un coefficient se trouve être important, son signe est codé et est orienté vers le LSP. L'algorithme examine ensuite la liste des ensembles et exécute une vérification d'amplitude de tous les coefficients de l'ensemble. Si un ensemble particulier se trouve être important il est alors divisé en sous ensembles et testé pour importance, autrement un seul bit est ajouté à la séquence de bits pour indiquer un ensemble insignifiant. Après qu'une passe au travers de LIS est achevée, une autre passe d'affinement au travers de LSP, excluant tous les coefficients ajoutés durant la passe de tri précédent, est initiée. Cette passe d'affinement s'accomplit en utilisant un codage progressif du plan de bits des coefficients ordonnés. Le seuil est par la suite diminué par un facteur de deux (résultant en une quantification uniforme des coefficients) et le processus entier est répété jusqu'à ce que le nombre de bits alloué s'épuise. Puisque toutes décisions raccordées faites par le codeur pendant qu'il cherche au travers des coefficients sont rajoutées à la séquence de bits, la situation des coefficients étant affinés ou classées n'est jamais explicitement transmise. le résultat de ce processus de tri affinement est ensuite codé. Le décodeur recrée les décisions du codeur pour reconstruire l'image.

2.5.3 Codage par Quantification Vectorielle

Les techniques de compression d'images exploitent généralement la redondance spatiale présente dans l'image. La quantification scalaire (QS) qui associe à une variable continue une variable discrète pouvant prendre un nombre plus faible, et fini de valeurs. Ces valeurs ne sont jamais totalement décorréélées, ou indépendantes.

Shannon a montré qu'il était toujours possible d'améliorer la compression de données en codant des vecteurs plutôt que des scalaires [27].

La quantification vectorielle (QV), développée par Gersho et Gray [28] a pris une place très importante dans le domaine de la compression d'images que ce soit dans le but de transmission ou d'archivage. La quantification vectorielle dans son sens le plus général est l'approximation d'un signal d'amplitude continue par un signal d'amplitude discrète. Elle peut être vue comme une application Q associant à chaque vecteur d'entrée x de dimension k un vecteur $y = Q(x)$ de même dimension appartenant à un ensemble fini Y appelé dictionnaire de taille finie N , $Y = (y_j, j=1 \dots N)$.

Le codage par QV consiste à trouver d'abord un dictionnaire constitué de vecteurs prédéfinis représentant le mieux l'image et se trouvant au niveau du codeur et du décodeur [13].

L'image est découpée en petits blocs carrés ou rectangulaires, généralement de taille égale à 4×4 ou 8×8 pixels. Chaque bloc étant transformé en vecteur x est comparé aux vecteurs du dictionnaire. L'indice ou l'adresse du vecteur le plus similaire au bloc à coder est transmis.

Le décodeur reconstruit facilement chaque bloc de l'image, en retrouvant pour chaque indice reçu le vecteur du dictionnaire qui lui correspond. La figure 2.7 illustre un schéma général d'un quantificateur vectoriel.

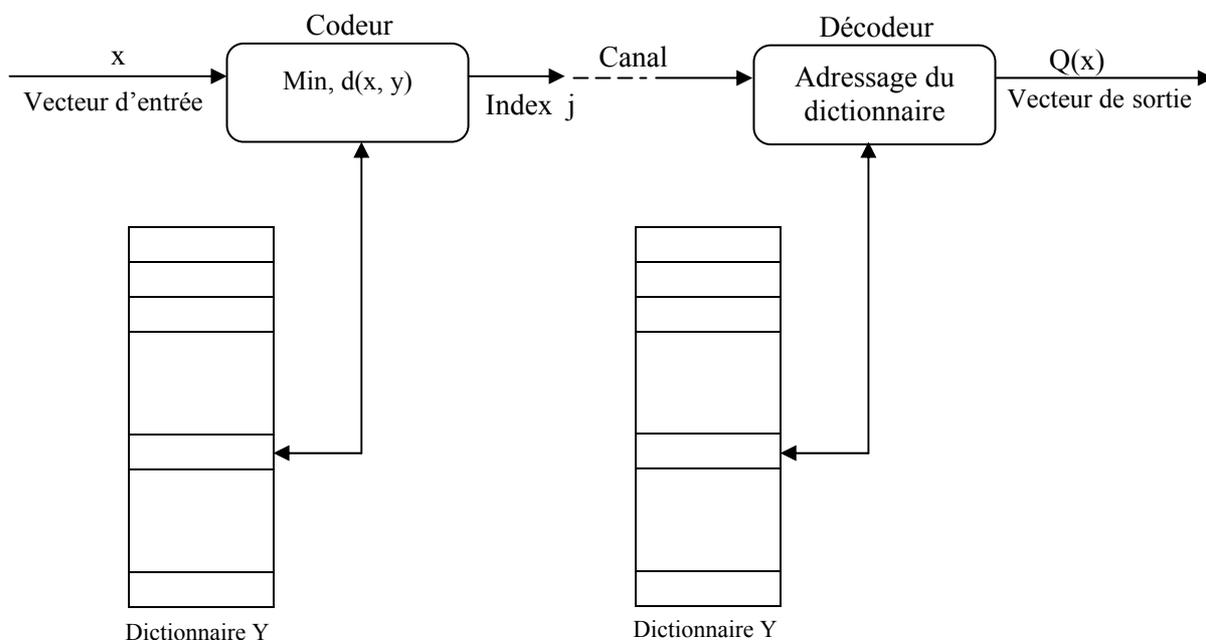


Figure 2. 7 : Principe du quantificateur vectoriel

Les algorithmes de construction du dictionnaire les plus utilisés sont ceux de Linde Buzo et Gray (LBG) [29]. La QV peut être combinée avec d'autres schémas de codage pour donner un codeur plus efficace. Cependant, la qualité du quantificateur vectoriel est conditionnée par l'algorithme de construction du dictionnaire. Celui-ci est défini par deux paramètres :

- Le nombre de vecteurs représentants N ,
- La taille des vecteurs k .

Les vecteurs représentent des blocs de l'image. Généralement, la taille du bloc est 4×4 ou 8×8 pixels.

2.5.4 La quantification scalaire

La quantification scalaire (QS) est une forme particulière de la (QV), celle où la dimension des vecteurs est égal à un. Elle consiste à quantifier les niveaux de gris des pixels les uns après les autres [30]. La figure 2.8 illustre la caractéristique en marche d'escalier du plus simple des quantificateurs scalaires, celui uniforme qui est entièrement déterminé par :

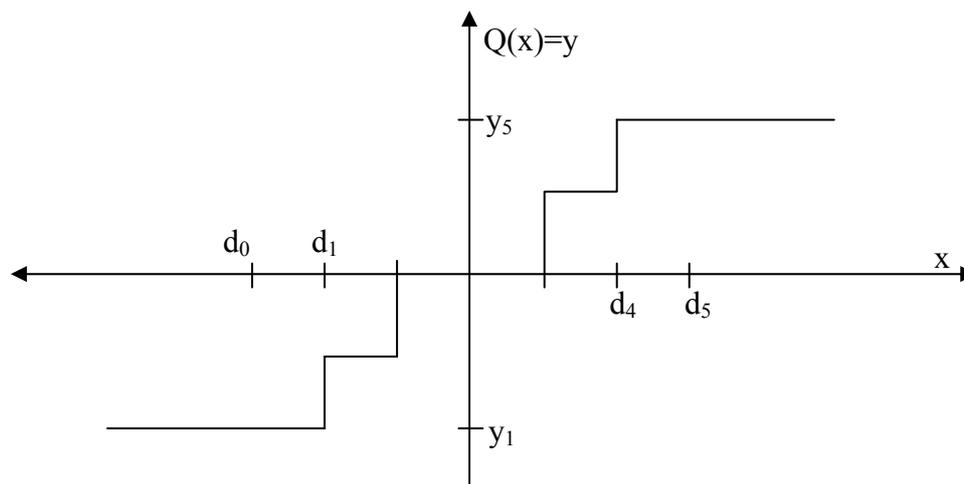


Figure 2.8 : Exemple de quantification scalaire uniforme pour $L=5$

Les $L+1$ niveaux de décisions : d_0, d_1, \dots, d_L qui partitionnent en L intervalles égaux l'axe des réels R et détermine le pas de quantification.

Les L valeurs de reproduction : y_0, y_1, \dots, y_L qui sont les centres de masses de chacun des intervalles de décision.

2.5.5 Méthode pyramidale

Ce type de codage est relativement plus récent que le codage transformé. Le codage en pyramide consiste en une description de l'image en un nombre de points de plus en plus réduit par filtrage passe – bas successifs. A la limite, toute l'image se réduirait en un point au sommet de la pyramide [7].

Soit F_0 une image donnée représentant la base ou le niveau zéro d'une pyramide. Le niveau n de la pyramide contient par filtrage passe-bas une version de F_0 dénotée F_1 dont la résolution est plus petite. Chaque pixel de l'image F_1 est une moyenne pondérée d'un ensemble de pixels avoisinants dans l'image F_0 . Ce processus est réitéré jusqu'à atteindre un certain niveau n de la pyramide. La seconde étape du codage consiste à générer une séquence d'images L_i ($i=0, \dots, n-1$) qui sont moins corrélés en comparaison avec les images F_i :

$$L_i(x, y) = F_i(x, y) - F'_{i+1}(x, y) \quad (2.3)$$

Où F'_{i+1} est une extension de l'image F_{i+1} obtenue par interpolation .

La dernière étape consiste à quantifier puis coder par entropie l'image F_n du niveau le plus élevé de la pyramide et la série d'images L_i . C'est à ce niveau que s'opère la compression.

2.5.6 Compression fractale

La méthode fractale est l'une des méthodes avec pertes utilisée pour la compression d'images fixes et animées [8][10]. La compression fractale développée par la société Iterated Function Systems (IFS) repose sur la constatation que toute image, avec un degré d'approximation plus ou moins grand, peut être représentée par des portions d'elle-même que

l'on modifie (déplacement, rotation, changement d'échelle...). L'algorithme consiste donc à décomposer une image en un immense « puzzle » avec des pièces de formes quelconques, et à rechercher les pièces qui sont similaires entre elles à une transformation affine près. Ce procédé est extrêmement long mais permet de réduire une image aux seules données qui décrivent les fonctions affines qui lient entre elles les pièces du puzzle.

La compression fractale devient une réalité opérationnelle avec l'introduction par Jacquin [8][30] de systèmes de fonctions itérées partitionnées (PIFS) qui diffèrent des systèmes de fonctions itérées classiques en ce sens que les transformations mises en jeu ne sont pas appliquées à l'image entière, mais à des sous parties de l'image, puisque celle-ci n'est pas fractale (image naturelle). Cette méthode constitue le modèle de référence pour le codage des images fixes et animées. Son principe effectue une mise en correspondance de blocs de l'image, par l'intermédiaire d'une partition avec une approche carré, en quadtree ou en rectangle (partition HV).

2.5.7 Algorithms adaptatifs de compression

Un nombre important de schémas adaptatifs ont été développés, mais la plupart sont des variantes des méthodes suivantes :

2.5.7.1 Sélection par seuillage

Cette méthode permet de sélectionner les coefficients supérieurs à un seuil donné [8]. Le nombre de coefficients sélectionnés étant différent d'un bloc à l'autre, fera varier le débit de sortie. En plus de la valeur du coefficient il faudra transmettre une information supplémentaire pour localiser les coefficients sélectionnés. Une alternative consiste par exemple à ne coder que les N plus grands coefficients du bloc afin de rendre le débit stable.

Un autre critère de sélection consiste à sélectionner par ordre croissant de la variance tous les coefficients supérieurs à un seuil prédéfini [7]. Cependant, l'information additionnelle de localisation des coefficients retenus est importante, à savoir du même ordre que l'information sur les valeurs de ces coefficients.

2.5.7.2 Sélection en pourcentage d'énergie

Une autre méthode de sélection des coefficients consiste à ordonner les n^2 coefficients du bloc transformé par ordre croissant sur les variances et ensuite déterminer le plus petit α pour lequel la quantité définie par [31] :

$$\mu = \frac{\sum_{u=0}^{\alpha} \sum_{v=0}^{\alpha} |G(u, v)|^2}{\sum_{u=0}^{n-1} \sum_{v=0}^{n-1} |G(u, v)|^2} \quad (2.4)$$

dépasse un seuil prédéfini (par exemple 0.99). Dans chaque bloc, les seuls éléments à coder seront alors les α coefficients et la valeur α . L'utilisation d'un seuil défini en fonction de la composante continue $G(0, 0)$ améliore la qualité de l'image.

2.5.8 Méthodes hybrides

Le terme hybride fait référence aux techniques qui combinent plusieurs méthodes, dans le but d'améliorer les performances et l'efficacité de la compression [19]. A titre d'exemple, la méthode hybride combinant le codage prédictif et le codage par transformée regroupe les avantages des deux techniques qui le composent, à savoir la simplicité de mise en œuvre de la prédiction et les performances de la transformée. L'application de cette méthode aux images fixes, revient à effectuer une transformation à une dimension le long des lignes et ensuite une MICD le long des colonnes. Dans le cas des images animées, on effectue une combinaison entre une transformation bidimensionnelle dans le domaine spatial et une MICD le long de la composante temporelle pour exploiter la redondance temporelle du signal d'image.

Cette technique est tout particulièrement intéressante. En effet, la norme " H261 " de compression des images animées stipulée par le CCITT fait partie de la classe générale des algorithmes hybrides à prédiction - transformation.

2.5.9 Les standards de compression d'images fixes

2.5.9.1 Le standards JPEG

Son nom (Joint Photographie Expert Groupe) provient de la réunion en 1982 d'un groupe d'experts de la photographie, dont le principal souci était de travailler sur les façons de transmettre des informations (images fixes ou animées) [16]. En 1986, L'IUT-T mis au point des méthodes de compression destinées à l'envoi de fax. Ces deux groupes se rassemblèrent pour créer un comité conjoint d'experts de la photographie (JPEG).

La norme JPEG est une norme de compression des images avec pertes [1]. Elle travaille sur des blocs de taille 8x8 pixels. Comme chaque point de chaque matrice image est une information codée sur 8 bits, il y a chaque fois 256 niveaux possibles (0-255). En soustrayant 128 à chaque élément, on met à zéro le milieu de la gamme de valeur possible : -128 à +127. Ce nombre permet de diminuer le poids de la matrice image, sans entraîner de modifications puisqu'il n'a qu'à signaler dans l'en-tête du fichier ce nombre à ajouter. Enfin chaque matrice image est partagée en blocs de 8x8. La DCT est donc effectuée sur chaque matrice 8x8 de valeurs de pixels, et elle donne une matrice 8x8 de coefficients de fréquence : l'élément (0, 0) représente la valeur moyenne du bloc, les autres indiquent la puissance spectrale pour chaque fréquence spatiale. Les coefficients les moins importants pour la restitution de l'image sont "amoindris" en divisant chaque élément par l'élément correspondant de la table de quantification. Ainsi les hautes fréquences s'atténuent rapidement. On remplace chaque premier élément (0, 0) des blocs (8x8) par sa différence avec l'élément correspondant du bloc précédent. Comme ces éléments sont les moyennes de leur bloc respectif, ils varient lentement. Ce seront donc de plus petits nombres qui prendront moins de place mémoire. On linéarise la matrice 8x8 en une suite de 64 nombres en s'arrangeant pour que les éléments les moins importants soit regroupés en fin de liste, on parle de méthode zig-zag. Ceci permettra par la suite d'améliorer encore plus la compression des données. Par la suite, ces coefficients sont compressés par un algorithme de Huffman sur la base de tables de codage prédéterminées.

2.5.9.2 La norme JPEG 2000

Le format JPEG qui est un standard largement répandu a cependant vieilli et présente un certain nombre de lacunes importantes pour le domaine de compression d'images : par forte compression, un artefact (effet mosaïque) très gênant, différences de couleurs, gestion de la transparence. C'est à partir de ce constat que le comité JPEG a ainsi décidé de créer une nouvelle norme JPEG 2000, intégrant des techniques plus performantes afin de répondre aux exigences formulées par les utilisateurs potentiels, en termes de performances attendues et de fonctionnalités souhaitées.

La norme JPEG2000 correspond en réalité à sept parties relativement indépendantes. Il est à noter qu'à ce jour, seule la première partie de JPEG2000 a atteint le stade de norme internationale [32]. Elle a été normalisée depuis le 2 janvier 2001. JPEG2000 définit déjà dans sa première partie, une grande majorité des applications touchant au domaine de l'imagerie numérique (archivage, diffusion sur Internet...). Cette norme utilise un procédé de décorrélation basé sur la transformation en ondelettes discrète (DWT) avec codage spécifique des sous images. Les données décorrélées sont quantifiées par un quantificateur scalaire uniforme. La fonction de quantification est défini par un pas de quantification de la sous-bande (chaque sous-bande peut avoir un pas de quantification différent), w est le coefficient d'ondelettes à quantifier et $Q(w)$ l'index de quantification .

JPEG 2000 code les index de quantification obtenus précédemment grâce à un codeur arithmétique adaptatif avec contexte. Les coefficients quantifiés sont préalablement groupés dans chaque sous-bande, en blocs rectangulaires (code-blocks) typiquement de taille 64x64 ou 32x32, puis chaque code-block est codé par un procédé de plan de bits par plan de bits, en commençant par les bits de poids le plus fort.

2.6 Conclusion

Le besoin de transmettre ou de stocker des données croît rapidement avec le développement des communications. Le recours à des techniques de réduction ou de compression de la masse énorme d'informations contenues dans les images joue un rôle essentiel, pour pallier aux problèmes de la capacité de stockage d'une part, et le temps de transmission d'autre part. Le chapitre suivant aborde la compression d'images par les algorithmes adaptatifs par DCT.

CHAPITRE 3 METHODOLOGIE ET RESULTATS

3.1. Introduction

L'image numérique est un signal bidimensionnel non stationnaire. En effet, la distribution énergétique varie d'une région à une autre [7]. Plusieurs travaux de compression des images fixes basés sur les statistiques de données de l'image ont été développés dans la littérature [2][3][4].

Dans ce chapitre, on s'intéresse à la compression avec pertes des images fixes données par les algorithmes adaptatifs de compression par DCT- 2D. Dans un premier temps, un algorithme de compression adaptatif de complexité moyenne connu sous le nom Adaptive Discrete Cosine Transform (ADCT) est développé. L'aspect fondamental d'optimisation est le critère de classification des blocs. Ce type d'adaptation consiste à construire un modèle de mesure de l'activité fréquentielle dans le bloc et ensuite adapter la sélection des coefficients.

Des modifications portées sur ce dernier ont donné naissance à un nouvel algorithme adaptatif de compression, appelé Modified Adaptive Discrete Cosine Transform (MADCT).

Comme extension et amélioration de la méthode adaptative MADCT, l'approche adaptative adoptée et implémentée AMADCT a été développée par la suite.

Après la présentation des approches adaptatives, et dans le but d'effectuer une étude comparative, un algorithme de compression sans classification basé sur la DCT est développé par la suite.

3.2. Algorithme de compression adaptative par la DCT-2D (ADCT)

Cette méthode basée sur la DCT proposée dans [2] a pour principe de base de diviser l'image en blocs de $(n \times n)$ pixels. Ces blocs sont transformés individuellement en utilisant la transformée en cosinus discrète (DCT-2D) pour produire en sortie un ensemble de n^2 coefficients (Figure 3.1).

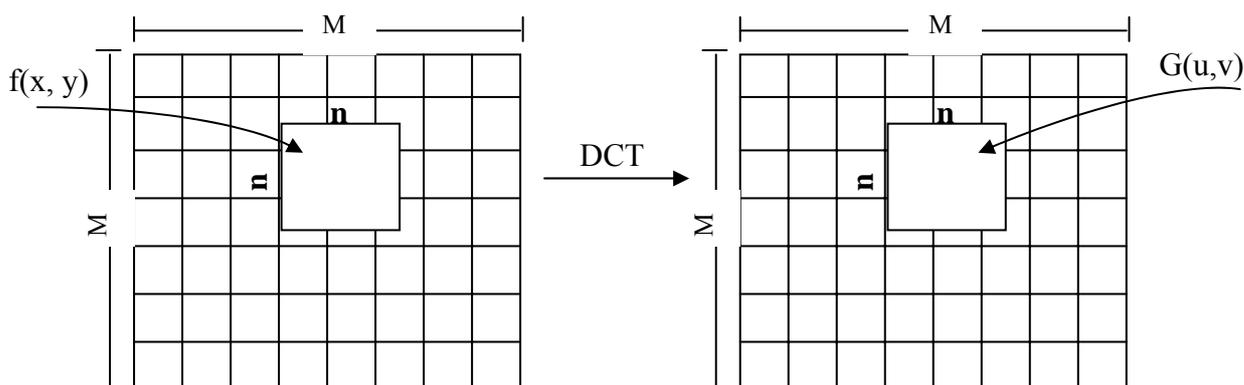


Figure 3.1: Transformation en DCT d'un bloc $n \times n$ pixels

Une classification des blocs transformés selon leurs niveaux d'activité est effectuée [3]. Le coefficient $G(0, 0)$ appelé composante continue (DC) caractérisant la moyenne des luminances des pixels est écarté du fait que les niveaux d'activités sont proportionnels à l'énergie des coefficients AC. L'énergie de AC dans le $(m, l)^{\text{ième}}$ bloc est donnée par la formule suivante :

$$E_{AC_{m,l}} = \sum_{u=0}^{n-1} \sum_{v=0}^{n-1} [G_{m,l}(u,v)]^2 - [G_{m,l}(0,0)]^2 \quad (3.1)$$

avec $m, l = 1, 2, \dots, M/n$ et M représente la dimension de l'image.

En se basant sur l'estimation de la distribution de la probabilité cumulative de l'énergie AC dans un bloc donné (Figure 3.2), l'énergie est classée dans quatre niveaux où chaque classe contient N blocs.

Avec :

$$N = \frac{1}{4} \left(\frac{M}{n} \right)^2 \quad (3.2)$$

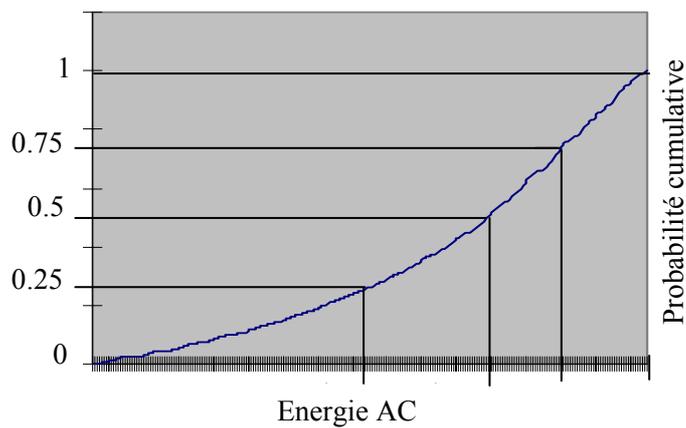


Figure 3.2 : Distribution de la probabilité de l'énergie AC dans un bloc

La figure 3.3 illustre le schéma du codeur ADCT .

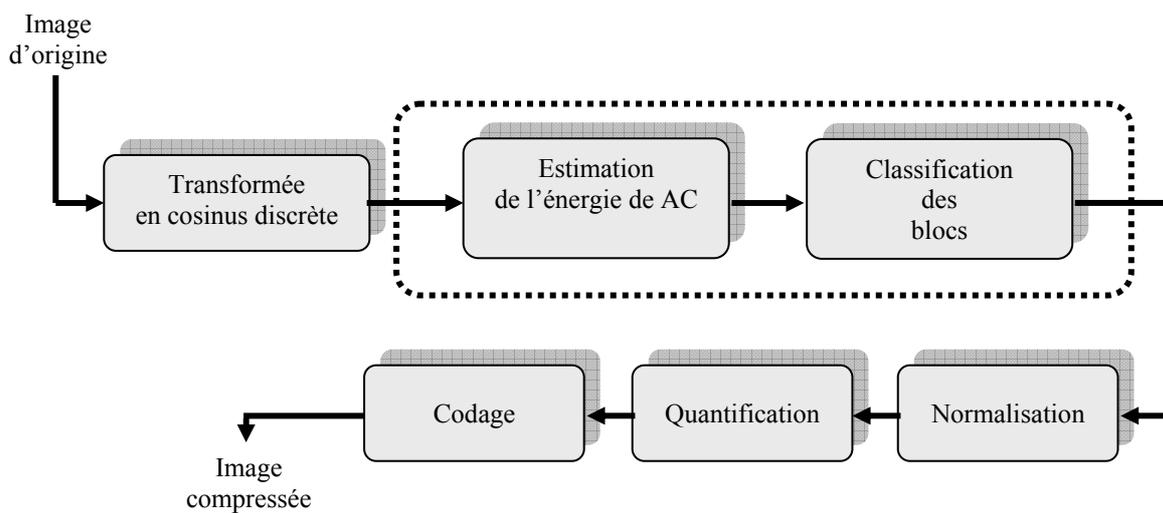


Figure 3.3 : Le schéma bloc de la ADCT

Après classification, pour chaque classe une matrice $n \times n$ dont chacun des éléments représente le nombre fixe de bits associé au coefficient correspondant dans le bloc transformé est défini. La construction de la matrice est basée sur la recherche des meilleurs représentants des coefficients. Afin de déterminer cette matrice, il est nécessaire de définir la variance. A partir de la distribution de l'énergie dans le domaine des fréquences cette méthode attribue d'avantage d'éléments binaires aux coefficients des basses fréquences qui transportent plus d'énergie (c'est-à-dire ayant des variances élevées). En revanche, les coefficients des hautes fréquences seront représentés par un nombre réduit d'éléments binaires. Il est important de signaler que la sélection des coefficients à transmettre est impliquée dans la distinction entre les éléments qui ont une affectation de quelques bits et ceux qui n'en ont aucune. En effet, les coefficients dont le nombre de bits égal à zéro sont écartés de la transmission. Par ailleurs, chaque coefficient transformé dans un bloc donné est normalisé en le divisant par sa variance, ceci afin de diminué l'étendue du domaine des valeurs. Tous les coefficients normalisés dans chaque classe sont par la suite quantifiés et codés.

3.3. Algorithme modifié de compression adaptative en DCT-2D (MADCT)

C'est une version modifiée de l'algorithme ADCT. Elle présente de meilleures performances par rapport à l'ADCT du point de vue du Rapport Signal / Bruit (PSNR) [4].

A la différence de l'ADCT, les blocs transformés sont classés suivant un critère de seuillage adaptatif dans n classes dont le nombre de blocs se diffère d'une classe à une autre. A partir des deux paramètres d'activité du bloc (énergie de AC (E_{AC}) et de DC (E_{DC})) un seuil d'activité des deux coefficients AC et DC est déterminé. Sa formule générale est donnée par :

$$T_{AC}(i) = \min_{E_{AC}} + \alpha_i \cdot \text{Gamme}_{E_{AC}}$$

$$T_{DC}(i) = \min_{E_{DC}} + \beta_i \cdot \text{Gamme}_{E_{DC}}$$
(3.3)

Avec

$$\text{Gamme}_{E_{AC}} = \max E_{AC} - \min E_{AC} \quad (3.4)$$

$$\text{Gamme}_{E_{DC}} = \max E_{DC} - \min E_{DC}$$

$$E_{AC_{m,l}} = \sum_{u=0}^{n-1} \sum_{v=0}^{n-1} [G_{m,l}(u,v)]^2 - [G_{m,l}(0,0)]^2$$

$$E_{DC_{m,l}} = [G_{m,l}(0,0)]^2 \quad (3.5)$$

Où

$m, l = 1, 2, \dots, M/n$.

i : représente l'indice de seuil.

α_i, β_i : variables estimées expérimentalement par l'opérateur tel que $0 < \alpha_i < 1$ et $0 < \beta_i < 1$.

Les fonctions min et max renvoient respectivement le minimum et le maximum de l'énergie des coefficients AC et DC pour tous les blocs.

Le schéma bloc des différentes étapes de l'algorithme MADCT est décrit dans la figure 3.4.

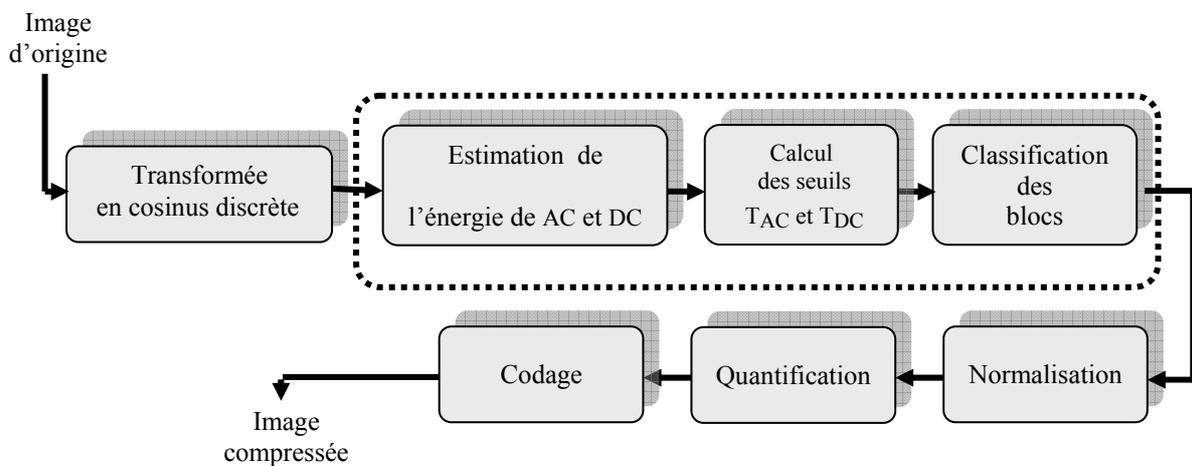


Figure 3.4 : Le schéma bloc de la MADCT

Les coefficients de la classe qui a la plus basse activité ne sont pas transmis. De ce fait, les coefficients sélectionnés appartenant aux autres classes sont normalisés, quantifiés et ensuite codés par une approche entropique.

3.4. Algorithme modifié et amélioré de compression adaptative en DCT-2D (AMADCT)

Notre méthode de compression proposée est inspirée de la technique MADCT. L'algorithme de base reste globalement inchangé. Ce schéma de compression est basé sur un critère de classification des blocs suivi d'une sélection des blocs ou des coefficients dans les différentes classes.

La figure 3.5 représente le diagramme du codeur de la méthode proposée. Les principaux blocs de ce schéma de codage sont :

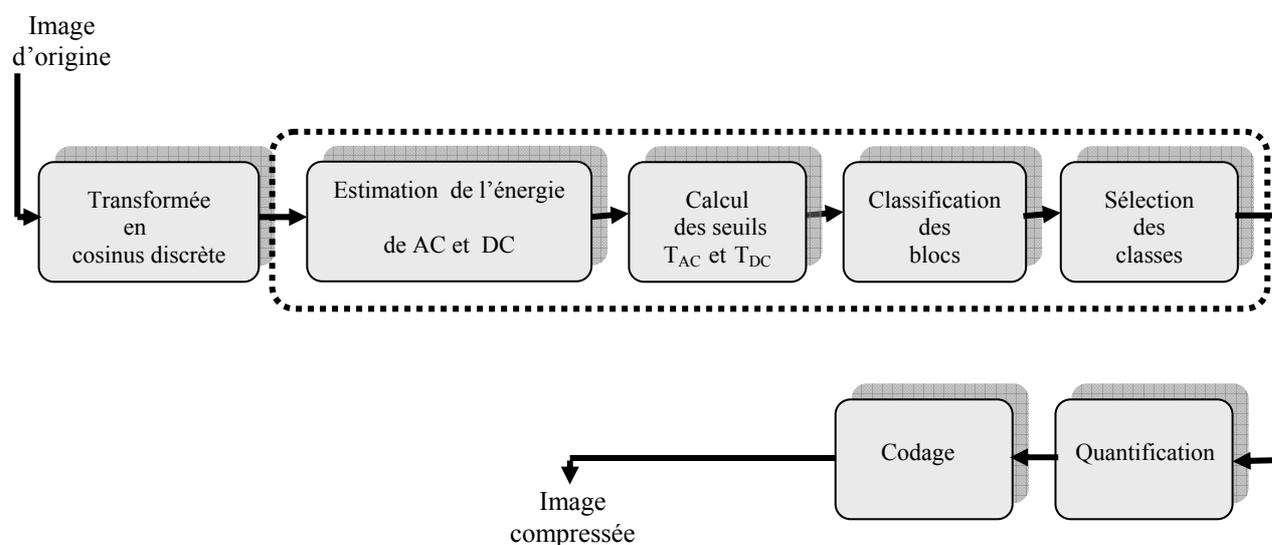


Figure 3.5 : Le schéma bloc de la MADCT améliorée

3.4.1. Transformée en cosinus discrète

Elle consiste, dans un premier temps, à décomposer l'image en blocs de taille 8x8 pixels. Dans un deuxième temps, chaque bloc est transformé par une DCT-2D. Cependant, avant d'effectuer la DCT sur un bloc de pixels, la valeur 2^{P-1} est retranchée de la valeur du pixel $f(x, y)$ afin de diminuer la dynamique de codage :

$$f^*(x, y) = f(x, y) - 2^{P-1} \quad (3.6)$$

pour $x, y = 0, 1, \dots, n-1$

P étant le nombre de bits utilisé pour coder un pixel de l'image.

Dans le cas où $P=8$, le décalage est évalué à 128 et nous obtenons des pixels dans une représentation signée :

$$f^*(x, y) = f(x, y) - 128 \quad (3.7)$$

Le nouvel intervalle de valeurs des pixels sera alors :

$$-127 \leq f^*(x, y) \leq +127 \quad (3.8)$$

L'opération de décalage inverse est nécessaire lors du décodage pour récupérer les pixels sous la forme non signée.

3.4.2. Classification des blocs

Tout comme la MADCT, la classification des blocs transformés se fait dans n classes, sur la base des énergies de coefficient AC (E_{AC}) et de coefficient DC (E_{DC}) et en respectant les seuils d'activités (T_{AC} et T_{DC}) définis respectivement dans l'équation (3.3).

Le nombre de classes dépend généralement du nombre relatif aux niveaux d'activité et du degré de changement d'activité dans l'image. Nous proposons de classifier les blocs transformés en trois classes à l'aide de la méthode développée dans [4]. Pour cela, deux seuils d'activité sont calculés :

$$T_{AC}(i) = \min_{E_{AC}} + \alpha_i \cdot \text{Gamme}_{E_{AC}} \quad i = 1$$

$$T_{DC}(i) = \min_{E_{DC}} + \beta_i \cdot \text{Gamme}_{E_{DC}} \quad i = 1$$
(3.9)

La stratégie adoptée pour la classification des blocs transformés dans les trois classes est la suivante :

Etant donné $E_{AC} < T_{AC}(1)$, un bloc est classé dans la classe 0 ou 1 si respectivement,

$$E_{DC} < T_{DC}(1) \quad (3.10)$$

ou

$$E_{DC} > T_{DC}(1) \quad (3.11)$$

Un bloc est classé dans la classe 2 si

$$E_{AC} > T_{AC}(1) \quad (3.12)$$

respectivement, pour toute valeur de E_{DC} .

3.4.3. Sélection des classes

Après la classification des blocs, chaque classe sera constituée d'un nombre différent de blocs. Plusieurs tentatives de sélection ont été proposées dans notre méthode, celle basée sur la sélection des blocs les plus significatifs pour chaque classe et autre sur la sélection des coefficients significatifs par l'intermédiaire d'un certain seuil prédéfini. Trois approches de sélection ont été implémentées et testées.

3.4.3.1. Sélection par la moyenne des énergies minimales par classe (MEM)

Dans cette méthode, la sélection des blocs pour chaque classe est guidée par l'énergie totale du bloc, grâce à un seuil fixé à l'avance (seuil prédéfini). Dans un premier temps, pour chaque classe, l'énergie totale portée par chaque bloc est calculée afin d'avoir l'énergie totale minimale appropriée à cette classe. A partir de cette dernière, la constante notée T désigne le seuil fixé pour toutes les classes, il est déterminé selon l'équation suivante :

$$T = \sum_{k=0}^{n-1} \frac{E_{T \min_k}}{n} \quad (3.13)$$

où, n étant le nombre de classes égal à 3 et $E_{T \min_k}$ dénote l'énergie totale minimale, correspondante à la classe k .

Un bloc d'une classe donnée est sélectionné si son énergie est supérieure au seuil T . Cependant, en plus des valeurs codées des coefficients des blocs retenus, il est nécessaire de transmettre pour le décodeur des informations de position pour localiser les blocs sélectionnés.

3.4.3.2. Sélection par la moyenne des valeurs Min - Max des coefficients (MVC)

Son principe, consiste à définir, à partir des valeurs minimale et maximale des coefficients AC et DC de la classe j ($j = 0, 1, 2$), deux seuils pour sélectionner les coefficients significatifs dans chaque bloc de la classe elle-même. La formule adoptée pour le calcul des seuils est donnée par l'expression suivante :

$$T_{AC_j} = \frac{Min_{AC_j} + Max_{AC_j}}{2}$$

$$T_{DC_j} = \frac{Min_{DC_j} + Max_{DC_j}}{2}$$
(3.14)

où, T_{AC_j} et T_{DC_j} dénotent les seuils de la $j^{\text{ième}}$ classe.

Tous les coefficients dont la valeur absolue est supérieure à la valeur du seuil qui lui correspond sont sélectionnés. En outre, l'information de positionnement des coefficients sélectionnés aussi bien que leur valeurs codées sont transmises.

3.4.3.3. Sélection par les valeurs des énergies minimales des coefficients AC-DC par classe (EMC)

Cette approche, opère une sélection des blocs sur une classe de la manière suivante : seuls les blocs portant une énergie E_{AC} et E_{DC} supérieure respectivement aux seuils définis dans l'équation (3.15) sont sélectionnés .

$$T_{AC_j} = \min \{ E_{AC_k} \} \quad (3.15)$$

$$T_{DC_j} = \min \{ E_{DC_k} \}$$

E_{AC_k} , E_{DC_k} , désignent respectivement l'énergie AC et DC du $k^{\text{ième}}$ bloc.

T_{AC_j} : seuil minimal correspond à la $j^{\text{ième}}$ classe ($j=1, \dots, n$).

Les blocs sélectionnés varient d'une classe à une autre en fonction de leur structure interne. Il faut alors tenir compte de l'information de positionnement des blocs sélectionnés en plus de l'information d'amplitude des coefficients appartenant à ces blocs.

Après différents tests comparatifs, cette approche de sélection a été adoptée et incorporée dans le codeur .

3.4.4. Quantification

Les coefficients des blocs obtenus de la phase de sélection sont injectés dans le quantificateur, chaque coefficient $G(u, v)$ est quantifié individuellement par un quantificateur uniforme. Le pas de quantification varie en fonction du rang (u, v) du coefficient dans le bloc transformé. Il s'acroît fortement quand (u, v) s'éloigne de $(0, 0)$. La dépendance fréquentielle reflète le fait que les coefficients des hautes fréquences sont moins fréquents que ceux situés aux basses fréquences et seront par conséquent moins finement quantifiés (pas de quantification élevé).

Le rôle de la quantification est d'éliminer l'information qui n'est pas visuellement perceptible. Les tables de quantification sont psychovisuelles : elles tiennent compte des caractéristiques visuelles de l'œil humain.

Le quantificateur uniforme est défini par l'équation suivante :

$$G_{q_{uv}} = \text{Round} \left(\frac{G_{uv}}{Q_{uv}} \right) \quad (3.16)$$

Où

$G_{q_{uv}}$: représente le coefficient quantifié,

G_{uv} : représente le coefficient transformé,

Q_{uv} : représente le pas de quantification associé,

Round [] : fonction qui donne l'entier le plus proche.

L'équation de déquantification est :

$$Gr_{uv} = G_{q_{uv}} \cdot Q_{uv} \quad (3.17)$$

où Gr_{uv} est le coefficient DCT reconstruit.

La table de la figure 3.6 est celle fournie dans la norme JPEG [16][21], son pas de quantification est beaucoup plus fin pour les coefficients correspondant aux basses fréquences.

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Figure 3.6 : Table des pas de quantification

Lors du processus de codage la table de quantification doit être transmise avec les données compressées ou fournie de l'extérieur pour le processus de décodage.

Les coefficients quantifiés font l'objet d'un prétraitement afin d'obtenir un codage entropique efficace. Ce prétraitement consiste à utiliser les techniques de balayage. Celles - ci cherchent à ordonner au maximum les coefficients quantifiés dans l'ordre décroissant de leurs énergies. Les techniques de balayage les plus utilisées sont : balayage en ZigZag, balayage en ligne par

ligne, balayage en colonne par colonne. Les différents balayages proposés ont pour but d'obtenir une séquence unidimensionnelle à partir d'un bloc de coefficients quantifiés.

Les performances obtenues par la méthode de balayage en ZigZag (Figure 3.7) nous ont amené à la choisir et à l'utiliser comme un chemin de parcours [7][23]. Elle est la plus efficace et la plus utilisée en codage entropique d'images.

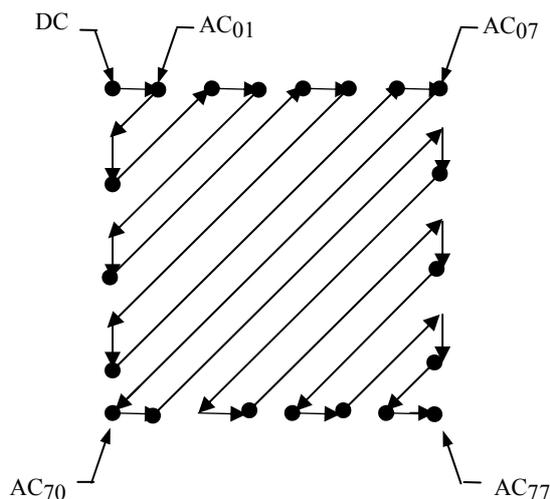


Figure 3.7 : Balayage en Zig-Zag

3.4.5. Codage

C'est la dernière étape de processus de codage MADCT amélioré. Les coefficients quantifiés sont codés par un algorithme de Huffman sur la base de tables de codage prédéterminées.

Les coefficients quantifiés présents à leur entrée au processus de codage sont exactement reconstruits à la sortie de décodeur au processus de décodage.

3.5. Décodage

Le Décodeur est considéré comme un récepteur chargé de la reconstruction de l'image. Le processus de décodage consiste à effectuer des traitements inverses au processus de codage (Figure 3.8). L'information complémentaire est par définition, l'information additionnelle à transmettre avec le flux de données compressées. Elle est nécessaire pour le décodage de

l'image. Pour pouvoir régénérer les blocs non sélectionnés à la réception, l'information de position des blocs sélectionnés doit être connue au niveau du décodeur. Cette information constitue alors l'information complémentaire.

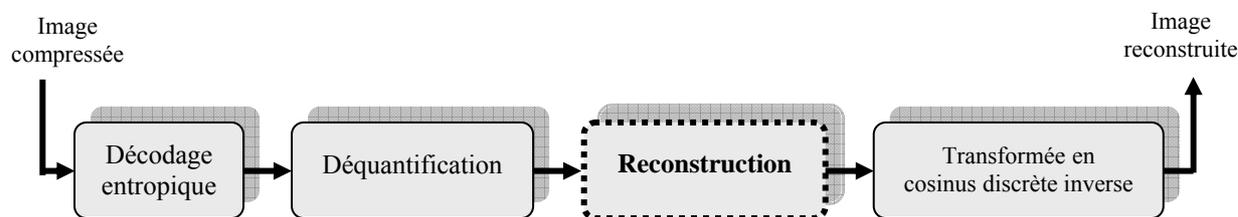


Figure 3.8 : Schéma de décodage

Une opération de reconstruction des blocs non sélectionnés dans le schéma de codage est incontournable afin d'éviter une dégradation de l'image reconstruite due au décalage des blocs. En effet, à chaque changement de ligne de blocs on a une accumulation de décalage. Elle est procédé en générale en deux étapes :

- 1- Repositionnement des blocs décalés;
- 2- Reconstruction de tous les blocs non sélectionnés.

Trois opérations de reconstruction ont été proposées.

3.5.1. Reconstruction par les coefficients du bloc précédent (CBP)

Elle consiste, à remplacer tous les blocs non sélectionnés par leurs précédents. Les coefficients d'un bloc de position $k \neq 0$ sont remplacé dans l'image par les coefficients du bloc précédent.

$$G_k(u, v) = G_{k-1}(u, v) \quad k \neq 0 \quad (3.18)$$

Si dans le schéma de codage le premier bloc de l'image ($k=0$) est non sélectionné alors la reconstruction de ce bloc se fait par les coefficients de premier bloc sélectionné par le codeur.

3.5.2. Reconstruction par la moyenne des coefficients (MOC)

Dans le même principe que l'approche précédente, la reconstruction de premier bloc de l'image ($k=0$) se fait par la moyenne et non par les coefficients de premier bloc sélectionné par le codeur. Par contre, pour la reconstruction d'un bloc de position $k \neq 0$ deux propositions sont données :

Si pour un bloc non sélectionné d'ordre k le bloc suivant d'ordre $k+1$ est sélectionné dans le schéma de codage alors les coefficients du bloc non sélectionné sont estimés comme la moyenne des coefficients des deux blocs voisins :

$$G_k(u, v) = \frac{G_{k-1}(u, v) + G_{k+1}(u, v)}{2} \quad (3.19)$$

Si non, le bloc est automatiquement rempli par la moyenne des coefficients du bloc précédent à savoir :

$$G_k(u, v) = M_{k-1}(u, v) \quad (3.20)$$

tel que M représente la moyenne du bloc précédent.

3.5.3. Reconstruction par la moyenne et les coefficients du bloc précédent (MOC-CBP)

Elle combine les deux approches précédentes. Dans cette technique, les blocs non sélectionnés de position $k \neq 0$ sont remplis par la moyenne des coefficients des deux blocs voisins. Sinon, en l'absence du bloc suivant, le bloc est rempli par les coefficients (et non la moyenne) du bloc précédent. Aussi, dans le cas où $k=0$, les coefficients de ce bloc seront remplacés par les coefficients du premier bloc sélectionné dans le schéma de codage.

Les tests effectués sur les différents types d'images (photographiques, médicales et satellitaires optiques et radar) montrent que la troisième approche (MOC - CBP) donne de meilleures performances.

En vue d'une étude comparative, un algorithme de compression sans classification est développé. L'algorithme proposée est très classique et très simple, consiste à effectuer une transformée en cosinus discrète sur des blocs de 8x8 pixels de l'image, à quantifier les coefficients obtenus en utilisant les tables de quantification fournies [20] et enfin à coder avec un codeur entropique de Huffman les coefficients quantifiés après réarrangement en Zig-Zag.

La figure 3.9 montre les blocs constituant le codeur. Les différents modules de ce codeur sont décrit ci dessous.

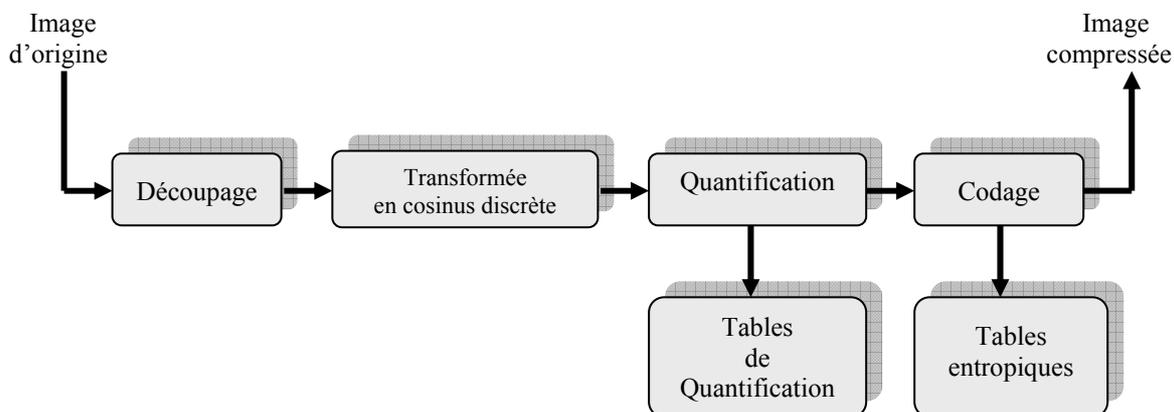


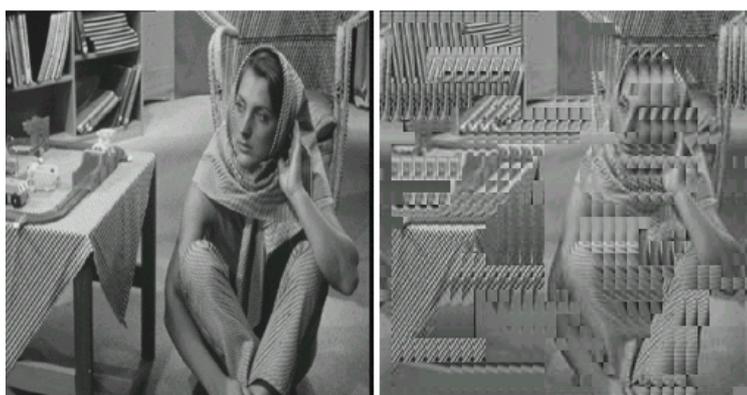
Figure 3.9 : Le schéma de codage

Le processus de décodage consiste à effectuer des traitements inverses au processus de codage, il produit une image reconstruite à partir des données d'image compressée. Les traitements effectués par le décodeur sont dans l'ordre : décodage entropique de Huffman, déquantification des coefficients obtenus et enfin une transformée en cosinus discrète inverse.

3.6 Tests et Résultats

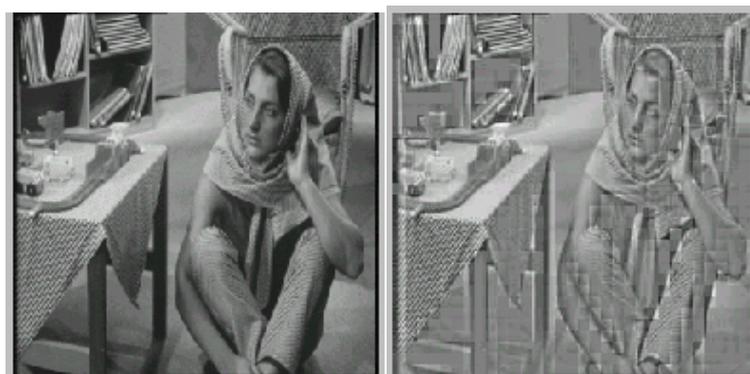
Pour tester notre algorithme, nous avons utilisé un ensemble d'images en niveaux de gris. Dans notre travail, nous avons choisi le C++ Builder 5.0 comme langage de programmation. Les tests ont été effectués sur différents types d'images (photographiques, médicales et satellitaires optiques et radar SAR (Synthetic Aperture Radar)). Le schéma du codage suivi est celui décrit ci-dessus en appliquant les trois approches de sélection. Quant au décodage, le processus inverse a été réalisé avec les trois différentes approches de remplissage des blocs non sélectionnés. Les résultats obtenus s'expriment en terme de taux de compression et du rapport signal à bruit (PSNR).

Les différents tests effectués sur différentes images en adoptant les deux approches de sélection MEM et MVC ont montré que les images reconstruites présentent un aspect visuel dégradé. En effet, les performances en terme de PSNR est de l'ordre 15.27 dB pour l'image Barbara (figure 3.10) dans le cas de l'approche MEM et l'image restituée est de qualité visuelle médiocre. Aussi, l'application de l'approche MVC sur les images Barbara et côte _d'azur_ comme l'illustre les figures 3.11 et 3.12 présentent des distorsions visibles à l'œil nu.



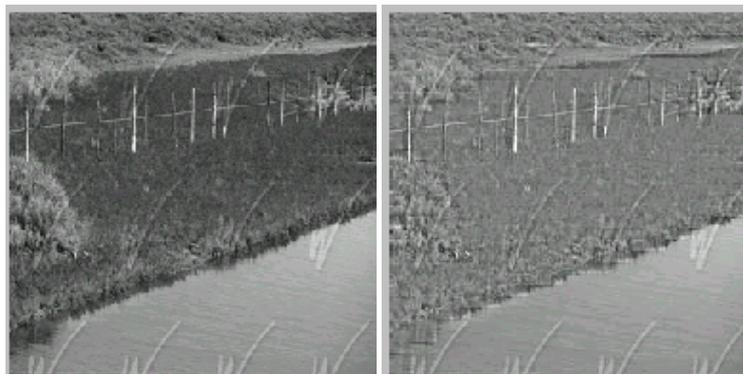
a) Image d'origine Barbara b) Image reconstruite

Figure 3.10 : Résultat de l'application de l'approche MEM pour l'image Barbara avec un débit 0.42 bpp (PSNR =15.27 dB)



a) Image d'origine Barbara b) Image reconstruite

Figure 3.11 : Résultat de l'application de l'approche MVC pour l'image Barbara avec un débit 0.80 bpp (PSNR =16.51 dB)



a) Image côte d'azur d'origine b) Image reconstruite

Figure 3.12 : Résultat de l'application de l'approche MVC pour l'image côte d'azur avec un débit 0.83 bpp (PSNR =18.01 dB)

Finalement, l'approche de sélection EMC est adoptée au codeur. Les premiers tests ont révélé que l'approche MOC est inefficace dans la reconstruction des images. En effet, les images reconstruites sont totalement dégradées.

De ce fait, les résultats que nous présentons se basent essentiellement sur les deux autres approches, en les comparant à chaque fois pour les trois types d'images photographiques, médicales et satellitaires optiques et radar SAR.

Les figures 3.13 à 3.16 montrent les résultats de l'application des deux approches sur les images satellitaires.

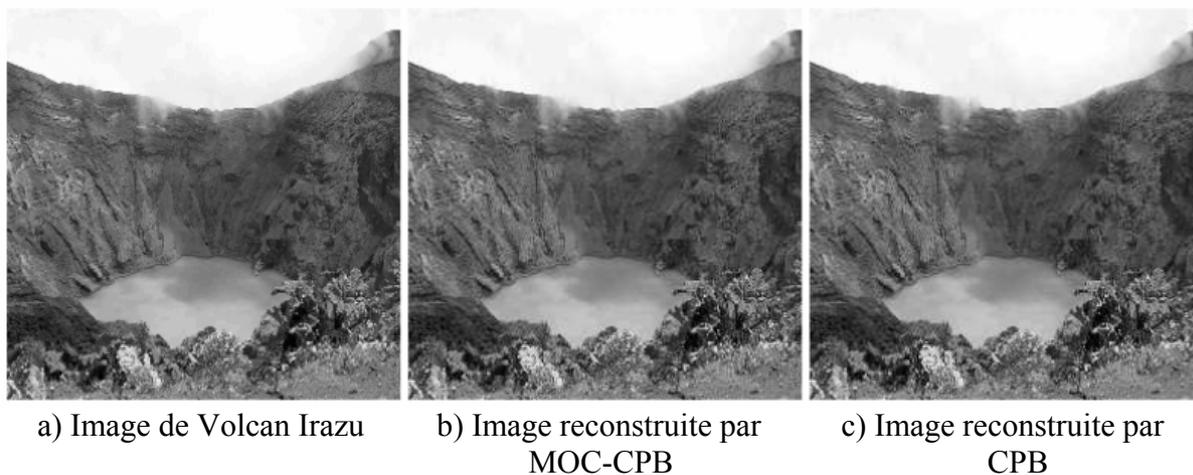


Figure 3.13 : Résultats de l'application des deux approches pour l'image Volcan _ Irazu avec l'approche MOC-CPB avec un débit 0.82 bpp (PSNR =26.83 dB) et avec l'approche CPB avec un débit 0.82 bpp (PSNR =26.70 dB)

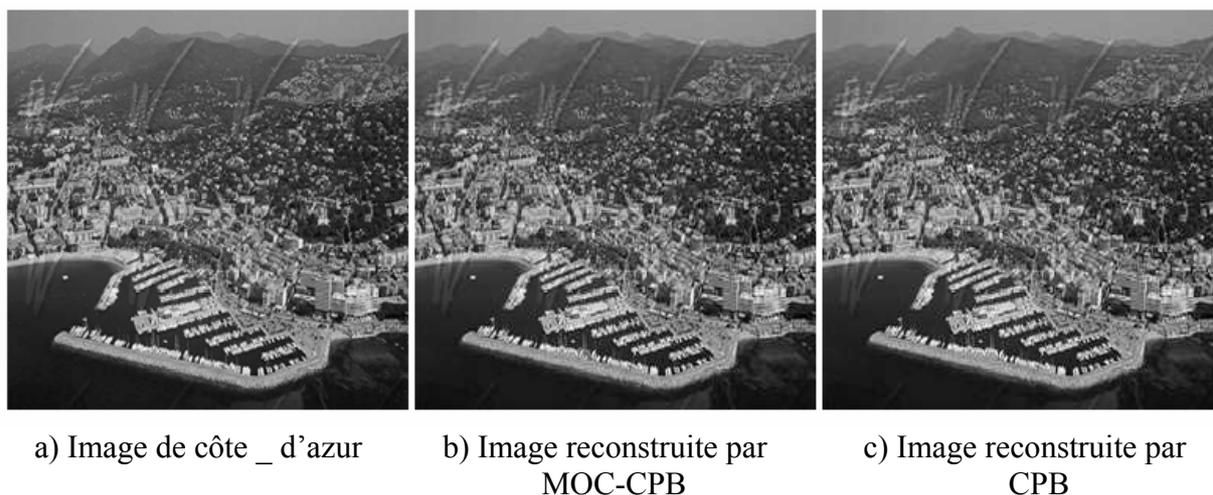


Figure 3.14 : Résultats de l'application des deux approches pour l'image côte _ d'azur avec l'approche MOC-CPB avec un débit 0.81 bpp (PSNR =27.77 dB) et avec l'approche CPB avec un débit 0.81 bpp (PSNR =27.49 dB)

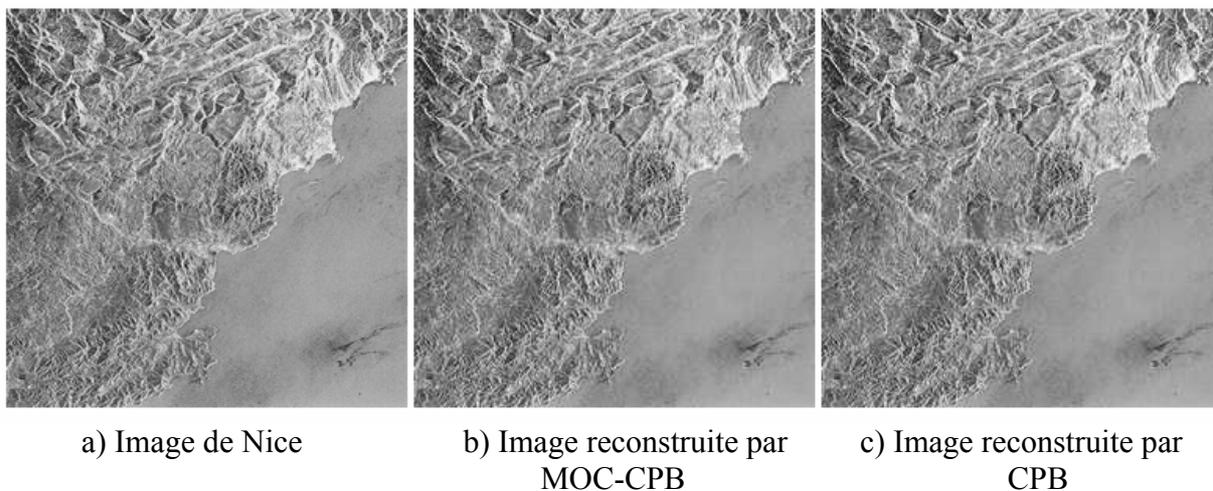


Figure 3.15 : Résultats de l'application des deux approches pour l'image Nice avec l'approche MOC-CPB avec un débit 0.86 bpp (PSNR =25.97 dB) et avec l'approche CPB avec un débit 0.86 bpp (PSNR =25.89 dB)

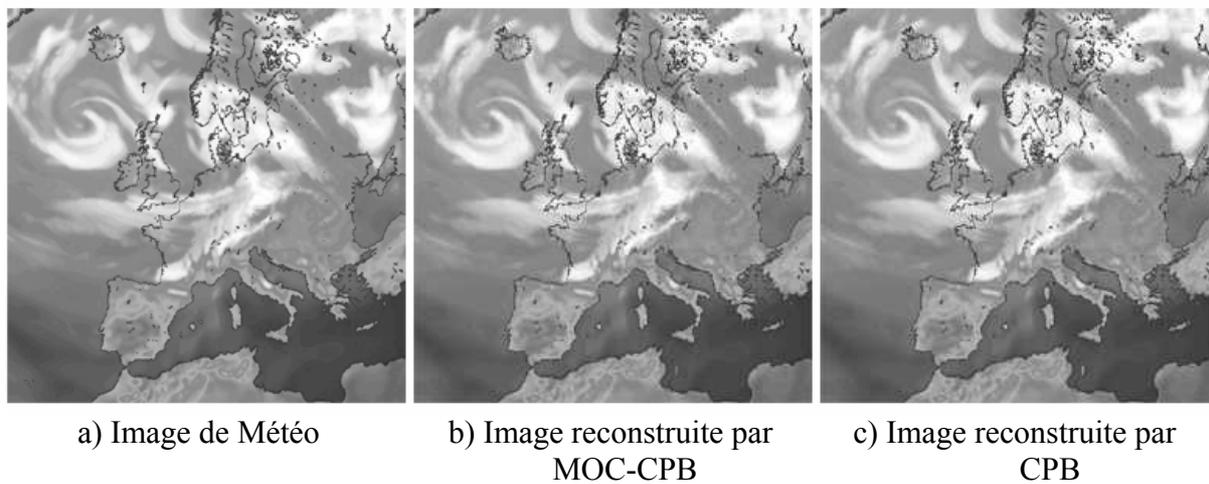


Figure 3.16 : Résultats de l'application des deux approches pour l'image Météo avec l'approche MOC-CPB avec un débit 0.80 bpp (PSNR = 30.24 dB) et avec l'approche CPB avec un débit 0.80 bpp (PSNR =30.13 dB)

Les tableaux 3.1 et 3.2 résument les performances obtenus sur les différents images satellitaires.

Images \ Performances	PSNR (dB)	Bit rate (bpp)
Volcan_Irazu	26.83	0.82
Côte_d'azur	27.77	0.81
Nice	25.97	0.86
Météo	30.24	0.80

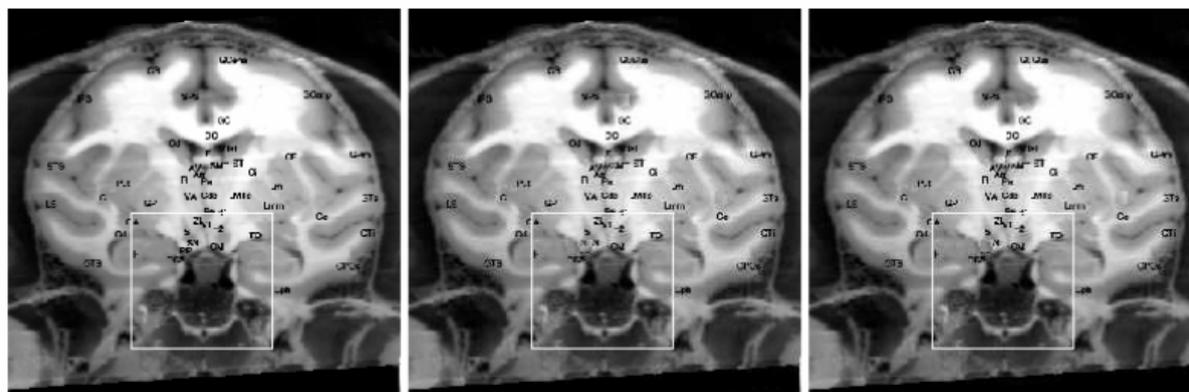
Tableau 3.1 : Performances de l'approche MOC-CBP

Images \ Performances	PSNR (dB)	Bit rate (bpp)
Volcan_Irazu	26.70	0.82
Côte_d'azur	27.49	0.81
Nice	25.89	0.86
Météo	30.13	0.80

Tableau 3.2 : Performances de l'approche CBP

Visuellement, les images restituées par ces deux approches sont de bonne qualité. Une légère amélioration en terme de PSNR est obtenue par l'approche MOC-CBP.

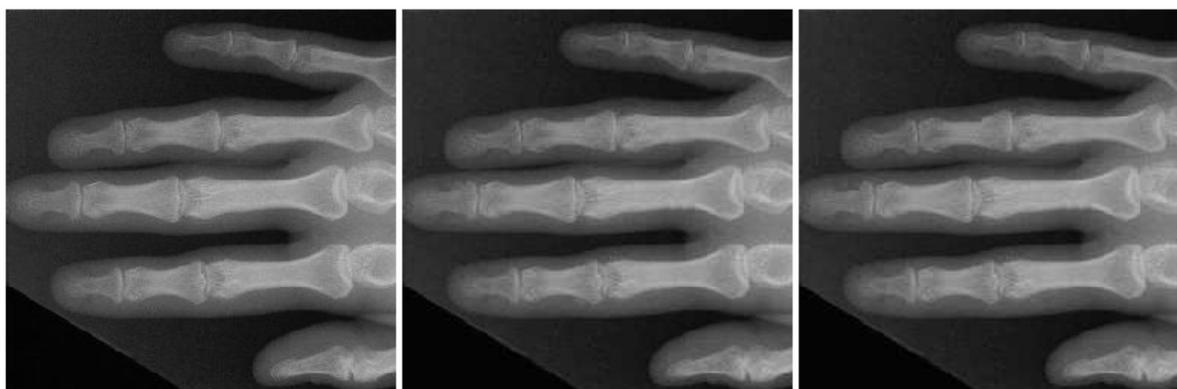
D'autres résultats de l'application des deux approches sur des images médicales (images neurologique crane et radiologique) sont donnés par les figures 3.17 et 3.18.



a) Image Neurologique

b) Image reconstruite par
MOC-CPBc) Image reconstruite par
CPB

Figure 3.17 : Résultats de l'application des deux approches pour l'image Neurologique avec l'approche MOC-CPB avec un débit 0.76 bpp (PSNR =29.99 dB) et avec l'approche CPB avec un débit 0.76 bpp (PSNR =29.99 dB)



a) Image Radiologique

b) Image reconstruite par
MOC-CPBc) Image reconstruite par
CPB

Figure 3.18 : Résultats de l'application des deux approches pour l'image radiologique avec l'approche MOC-CPB avec un débit 0.77 bpp (PSNR =37.57 dB) avec l'approche CPB avec un débit 0.77 bpp (PSNR =37.50 dB)

Les tableaux 3.3 et 3.4 résument les résultats de l'application des deux approches.

Performances Images	PSNR (dB)	Bit rate (bpp)
Neurologique	29.99	0.76
Radiologique	37.57	0.77

Tableau 3.3 : Performances de l'approche MOC-CBP

Performances Images	PSNR (dB)	Bit rate (bpp)
Neurologique	29.99	0.76
Radiologique	37.50	0.77

Tableau 3.4 : Performances de l'approche CBP

Les résultats obtenus par l'approche MOC-CBP pour les même taux de compression sont proches de ceux produits par l'approche CBP. On peut aussi noter dans ce type d'images que l'apparence visuelle de ces images par l'approche MOC-CBP est de bonne qualité que celle de l'approche CBP.

Afin de compléter cette série de résultats, les figures 3.19, 3.20 et 3.21 donnent les résultats de compression sur les images photographiques par les deux approches précédentes.

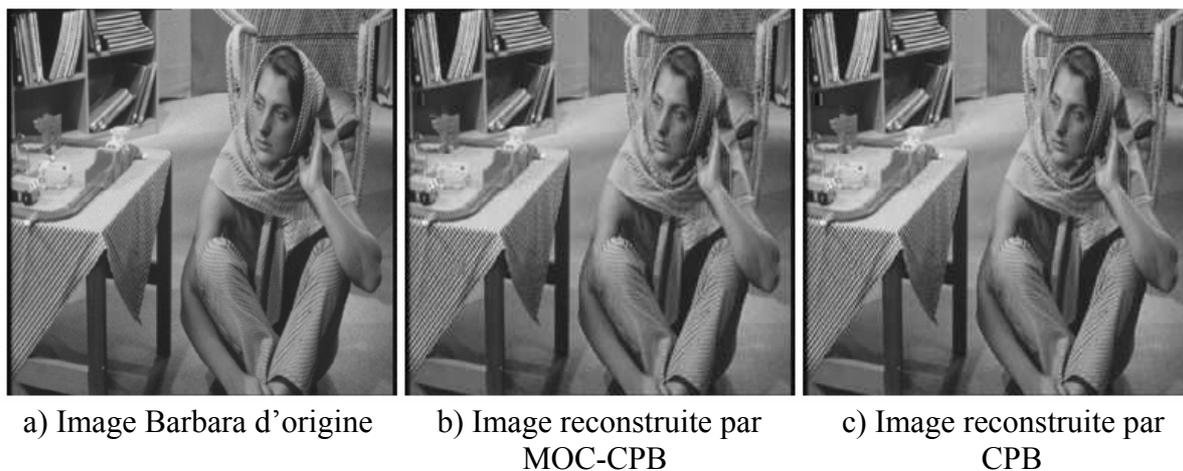


Figure 3.19 : Résultats de l'application des deux approches pour l'image Barbara avec l'approche MOC-CPB avec un débit 0.83 bpp (PSNR = 31.30 dB) et avec l'approche CPB avec un débit 0.83 bpp (PSNR = 31.06 dB)

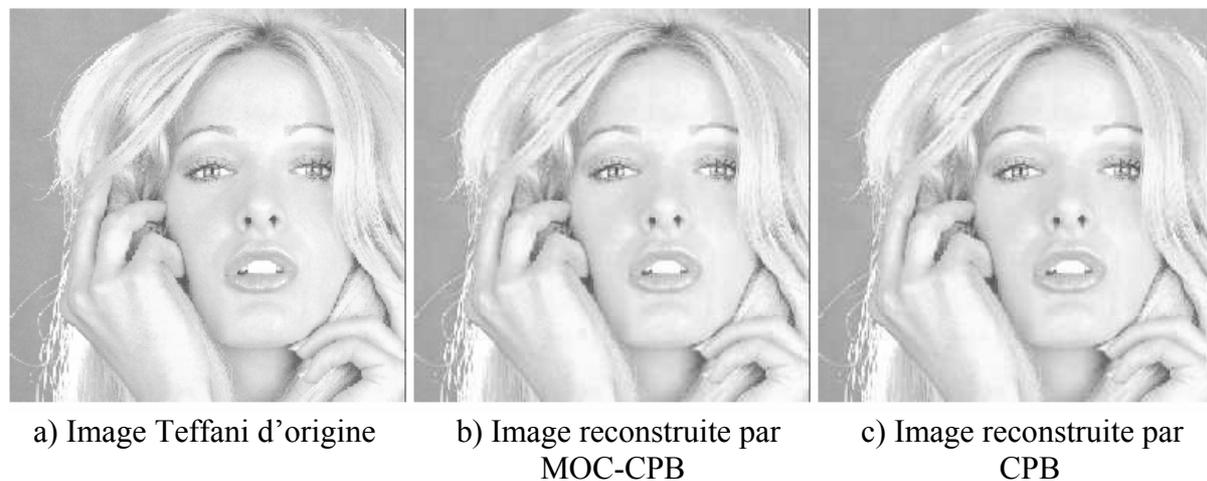


Figure 3.20 : Résultats de l'application des deux approches pour l'image Teffani avec l'approche MOC-CPB avec un débit 0.79 bpp (PSNR = 32.43 dB) et avec l'approche CPB avec un débit 0.79 bpp (PSNR = 32.39 dB)

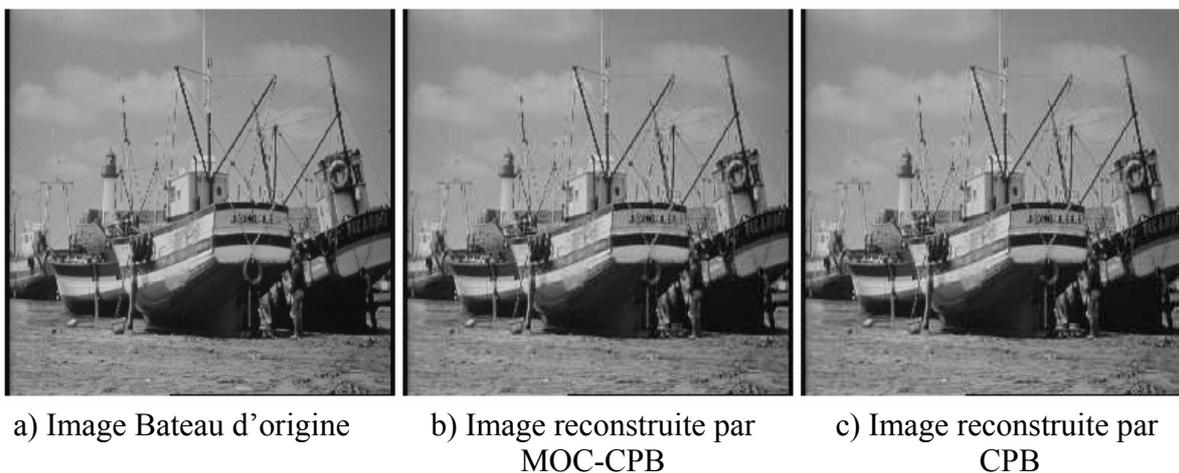


Figure 3.21 : Résultats de l'application des deux approches pour l'image Bateau avec l'approche MOC-CPB avec un débit 0.80 bpp (PSNR =30.98 dB) et avec l'approche CPB avec un débit 0.80 bpp (PSNR =30.75 dB)

Les tableaux 3.5 et 3.6 donnent des indications sur les résultats des figures 3.19, 3.20 et 3.21.

Images \ Performances	PSNR (dB)	Bit rate (bpp)
Barbara	31.30	0.83
Teffani	32.43	0.79
Bateau	30.98	0.80

Tableau 3.5 : Performances de l'approche MOC-CPB

Images \ Performances	PSNR (dB)	Bit rate (bpp)
Barbara	31.06	0.83
Teffani	32.39	0.79
Bateau	30.75	0.80

Tableau 3.6 : Performances de l'approche CBP

Numériquement les tableaux 3.5 et 3.6 montrent que l'approche MOC-CBP présente une nette amélioration par rapport à l'approche CBP en termes de PSNR. Nous constatons également que la qualité visuelle des images reconstruite avec les deux approches est aussi bonne que les autres types d'images.

D'après les résultats du PSNR donnés par les tableaux 3.1 à 3.6, nous pouvons conclure que l'approche MOC-CBP offre de bonnes performances par rapport à l'approche CBP.

Par ailleurs, nous avons effectué une comparaison entre l'approche classique « sans classification » et celle que nous avons implémentée. Les figure 3.22, 3.23 et 3.24 montrent les résultats de l'application des deux approches avec et sans classification sur les trois différentes images (Volcan _ Irazu, Neurologique et Barabra).

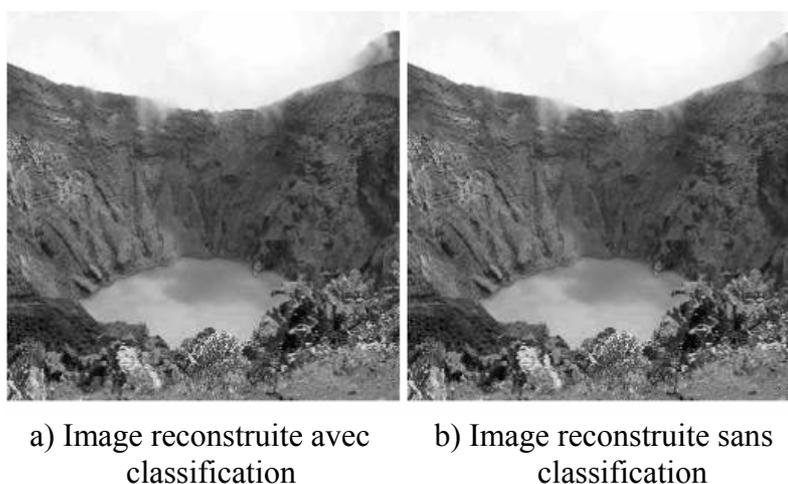


Figure 3.22 : Résultats de l'application des deux approches pour l'image Volcan _ Irazu reconstruite avec classification avec un débit 0.82 bpp (PSNR=26.83 dB) et sans classification avec un débit 0.84 bpp (PSNR =27.15 dB)

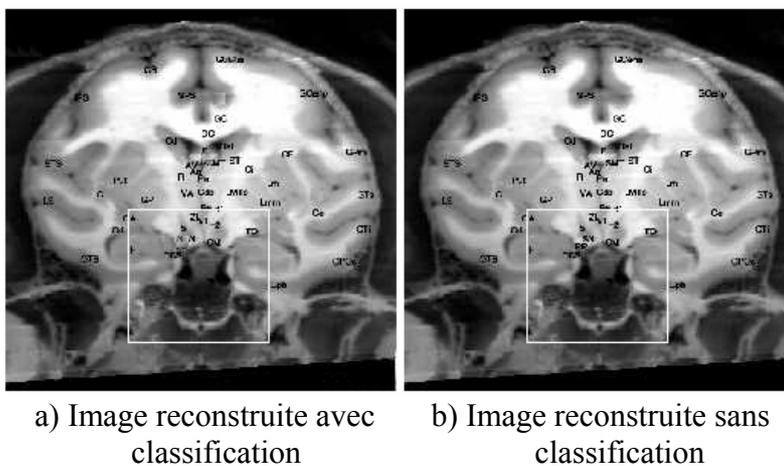


Figure 3.23 : Résultats de l'application des deux approches pour l'image Neurologique reconstruite avec classification avec un débit 0.76 bpp (PSNR=29.99 dB) et sans classification avec un débit 0.81 bpp (PSNR =32.06 dB)

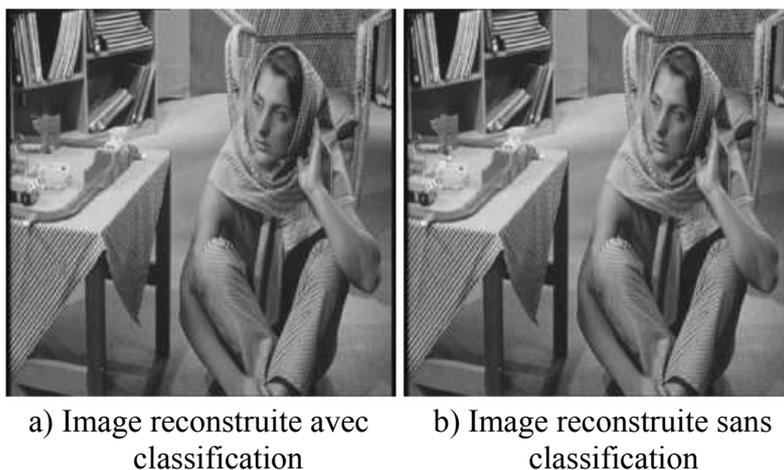


Figure 3.24 : Résultats de l'application des deux approches pour l'image Barabra reconstruite avec classification avec un débit 0.83 bpp (PSNR=31.30 dB) et sans classification avec un débit 0.84 bpp (PSNR =31.54 dB)

Les résultats fournis par notre approche sur ses différentes images présentent une nette amélioration en terme de taux de compression par rapport à l'approche classique. Il convient toutefois de signaler que l'image reconstruite par notre approche est toujours visuellement de bonne qualité.

3.7 Conclusion

L'approche MOC-CBP qui est une combinaison des deux approches citées plus haut (voir paragraphe 3.4.3) offre une bonne qualité de reconstruction de l'image.

CONCLUSION

Le domaine de la compression est l'objet d'efforts intenses et continuels pour la recherche et la conception de méthodes permettant une réduction substantielle de la quantité de données dans la représentation numérique des images.

Le projet consiste à développer un système de compression des images fixes, cela en vue d'une transmission ou d'un stockage efficace. La technique mise au point est basée sur le principe de transformation cosinus discrète (TCD) où l'information de l'image est représentée sous une forme moins redondante et plus condensée.

Dans un premier temps, nous avons répertorié les différentes techniques de compression, qui se sont succédées au fil du temps. Tout en précisant, le besoin en termes de compression fixes, de supports de stockage et de transmission, qui se fait ressentir sans cesse. De ce fait, les efforts des chercheurs du domaine ont permis le développement d'un grand nombre de méthodes de compression tels que les méthodes de compression adaptatives dans le plan transformé DCT .

La transformation cosinus possède les caractéristiques essentielles pour réaliser une compression efficace. En effet, elle conduit à une décorrélation importante des pixels dans l'image. Cette décorrélation se traduit par des propriétés de concentration de l'information sur un nombre réduit de coefficients dans le domaine transformé.

Dans le plan transformé, une classification et une sélection des blocs fondées sur un critère de seuillage sont effectuées. Les coefficients des blocs sélectionnés font l'objet d'une quantification uniforme, ensuite d'un codage entropique.

Arrivés au terme de notre projet, nous estimons avoir atteint l'essentiel de notre objectif qui rappelle le, est la réalisation d'un système de compression adaptatif basé sur la transformée en cosinus discrète (AMADCT).

Pour ce qui est des perspectives, une première voie consiste à adapter le codeur aux images couleurs. Aussi, la reconstruction du bloc non sélectionné au décodeur peut être fait à partir de ses blocs voisins.

APPENDICE A

Annexe 1

1 Introduction

De nos jours, l'image est devenue un support physique d'information privilégié. Traiter une image, c'est essentiellement l'analyser, lui faire subir des opérations pour la transmettre ou la stocker.

L'image est apparue d'abord, du point de vue scientifique, comme un outil de connaissance essentiel et plus récemment, comme une technologie innovante et performante dans les secteurs industriels de la production automatisée. Cet intérêt amplifié par les possibilités croissantes de l'usage des images est lié fondamentalement au fait que chez l'homme, la vision est le sens perceptif dominant (près de 60% du cortex cérébral humain est utilisé par le système visuel).

2 Définition de l'image

Parmi les nombreuses définitions du mot image, on peut retenir celle qui la définit comme étant la représentation exacte, ou la reproduction analogique d'une scène réelle, d'un être ou d'une chose par la peinture, la sculpture, le dessin, la photographie, le film,....etc. Mathématiquement, l'image est décrite par une fonction $F(x,y)$ définie dans un domaine borné; tel que x et y sont les coordonnées spatiales d'un point de l'image et F est une fonction d'intensité lumineuse et de couleur. L'image analogique est inexploitable par l'ordinateur ce qui nécessite sa numérisation.

3 Image numérique

L'image numérique est celle dont la surface est divisée en éléments de tailles fixes appelés cellules ou pixels, ayant chacun comme caractéristique un niveau de gris ou des

couleurs prélevés à l'emplacement correspondant dans l'image réelle, ou calculés à partir d'une description interne de la scène à représenter.

La numérisation d'une image est la conversion de celle-ci de son état analogique qui est une distribution continue d'intensité lumineuse dans un plan XOY en une matrice bidimensionnelle de valeurs numériques $f(x, y)$ où (x, y) sont les coordonnées cartésiennes d'un point de l'image, $f(x, y)$ est le niveau de gris en ce point .

Pour des raisons de commodité de représentation pour l'affichage et l'adressage, les données images sont généralement rangées sous formes de tableau I de n lignes et p colonnes. Chaque élément $I(x, y)$ représente un pixel de l'image et à sa valeur est associé un niveau de gris codé sur m bits (2^m niveaux de gris; 0 = noir ; 2^{m-1} = blanc). La valeur en chaque point exprime la mesure d'intensité lumineuse perçue par le capteur.

L'image peut être considérée comme la représentation de la distribution en 2-D de l'intensité d'un phénomène ou comme un ensemble de formes en 3-D où l'intensité du signal génère la surface du volume.

4 Caractéristiques d'une image numérique

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

Pixel

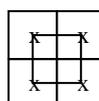
Le pixel (contraction de picture element) est aussi appelé tâche élémentaire (task), selon le vocabulaire de télédétection aérospatiale :

- Un pixel est la « plus petite surface homogène constitutive d'une image enregistrée, définie par les dimensions de la maille d'échantillonnage.

Un pixel et son barycentre



Une maille d'échantillonnage



- Le pixel est l'unité de résolution d'une image numérique : il correspond au nœud (x, y) de la grille définie par X*Y, où X est le nombre de colonnes et Y, est le nombre de lignes.

Dimension du pixel

La dimension théorique du pixel varie selon les systèmes d'acquisition des données : de 10*10 m pour l'image SPOT en mode panchromatique jusqu'à 5*5 Km pour l'image Météosat en passant par toute une gamme de dimension intermédiaire : par exemple, 20m pour les images SPOT multispectrales, 30m pour les images Landsat TM, 80m pour les images Landsat MSS.

Attribut du pixel

Un pixel a un attribut (une valeur) qui peut avoir diverses significations :

- o Une valeur de niveau de gris (radiométrie) : c'est le cas par exemple d'une photographie aérienne numérisée ou de l'imagerie satellitaire. Une valeur de niveau de gris correspond à la numérisation de la luminance c'est à dire la quantité d'énergie émise ou réfléchi, par une ou plusieurs surfaces géographiques dans le cadre d'une tâche élémentaire vers un capteur. Après numérisation, la luminance devient une valeur radiométrique codée. En règle générale, le nombre de niveaux de gris dépend du nombre de bits utilisés pour décrire chaque pixel de l'image. Plus ce nombre est important, plus les niveaux possibles sont nombreux. Par exemple, la luminance est numérisée sur 256 niveaux (codage sur 8 bits, soit 2^8 possibilités de codage); on trouve aussi sur 10 bits (1024 niveaux de gris) et même sur 16 bits (65536 niveaux de gris) en imagerie radar ou en imagerie thermique. Les valeurs radiométriques codées se réfèrent à l'origine zéro qui correspond à la valeur de luminance minimale de l'image.
- o Une valeur d'altitude : un Modèle Numérique de Terrain (MNT) représente une surface topographique sur laquelle chaque pixel reçoit une valeur d'altitude en mètre, décimètre ou centimètre.

Dimension

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

Résolution

C'est la clarté ou la finesse de détails atteinte par un moniteur ou une imprimante dans la production d'images. 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 désigner le nombre total de pixels affichables horizontalement ou verticalement sur un moniteur, plus grand est ce nombre, meilleure est la résolution.

La définition la plus générale de la résolution d'une image numérique est le nombre de pixels par pouce (dots per inch). Selon que le rapport pixels / pouce soit plus ou moins grand, la résolution est plus ou moins élevée et les détails de l'image plus ou moins apparents. Donc plus la résolution de l'image est haute, plus l'image est de meilleure qualité .

Bruit

Le bruit est à l'origine d'un grand nombre de difficultés lorsque l'on désire analyser un quelconque signal. L'image ne fait pas exception à cette règle. Il est considéré comme une perturbation parasite et aléatoire qui se superpose au signal et qui altère la qualité.

Un bruit (parasite) dans une image est un phénomène de brusque variation de l'intensité d'un pixel par rapport à ses voisins. Il provient généralement de l'éclairage des dispositifs optiques et électroniques du capteur.

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.

Il permet de donner un grand nombre d'information sur la distribution des niveaux de gris (ou couleur) et de localiser entre quelles bornes est répartie la majorité des niveaux de gris (ou couleur) dans une image .

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

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. Pour représenter les images en niveau de gris, on peut attribuer à chaque pixel de l'image une valeur correspondant à la qualité de lumière renvoyée. Cette valeur est comprise entre 0 et 255. Chaque pixel n'est donc plus représenté sur un bit comme dans les images monochromes, mais sur un octet. Pour cela, il faut que le matériel utilisé pour afficher l'image soit capable de produire les différents niveaux de gris correspondants.

Images couleurs

Même s'il est parfois utile de pouvoir représenter des images en noir et blanc, les applications multimédias utilisent le plus souvent des images en couleurs. La représentation des couleurs s'effectue de la même manière que celle des images monochromes avec cependant quelques particularités. En effet, il faut tout d'abord choisir un modèle de représentation. 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 RGB).

- La représentation en couleurs réelles

Elle consiste à utiliser 24 bits pour chaque point de l'image. Huit bits sont employés pour décrire la composante rouge (R), huit pour le vert (V) et huit pour le bleu (B). Il est ainsi possible de représenter environ 16.7 millions de couleurs différentes simultanément. Cela est cependant théorique, car aucun écran n'est capable d'afficher 16 millions de points. Dans la plus haute résolution (1600 x 1200), l'écran n'affiche que 1 920 000 points. Par ailleurs, l'œil humain n'est pas capable de distinguer autant de couleurs.

- La représentation en couleurs indexées

Son principe consiste à déterminer le nombre de différentes couleurs utilisées dans l'image, puis à créer une table de ces couleurs en attribuant à chacune une valeur numérique correspondant à sa position dans la table. La table appelée palette, comporte également la description de chacune des couleurs, sur 24 bits.

- Autres modèles de représentation

Le modèle RGB représentant toutes les couleurs par l'addition de trois composantes fondamentales, n'est pas le seul possible. Il en existe de nombreux autres. L'un d'eux est particulièrement important. Il consiste à séparer les informations de couleurs (chrominance) et les informations d'intensité lumineuse (luminance). Il s'agit du principe employé pour les enregistrements vidéo. La chrominance est représentée par deux valeurs (selon des modèles divers) et la luminance par une valeur.

5 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 un signal numérique. Elle dépend de :

- 1) La qualité des périphériques de numérisation de l'image, du nombre de niveaux de gris ou de couleurs enregistrées, etc.
- 2) La qualité de l'affichage à l'écran : définition de la carte graphique, nombre de teintes disponibles simultanément, calibrage de l'écran, etc.

Les critères d'appréciation de la qualité d'une image, tels que cités succinctement ci-dessus, dépendent largement de la structure même de l'image réaliste ou conceptuelle et de son mode de représentation (bitmap ou vectorielle).

6 Images bitmap et images vectorielles

On distingue généralement deux grandes catégories d'images :

- Les images bitmap : il s'agit d'images pixellisées ou matricielles, c'est-à-dire un ensemble de « points » .
- Les images vectorielles : les images vectorielles sont des représentations d'entités géométriques telles qu'un cercle, un rectangle, ellipse, arc ou un segment. Ceux-ci sont représentés mathématiquement par un nombre de points limité (un rectangle est défini par deux points). C'est le processeur qui sera chargé de « traduire » ces formes en informations interprétables par la carte graphique.

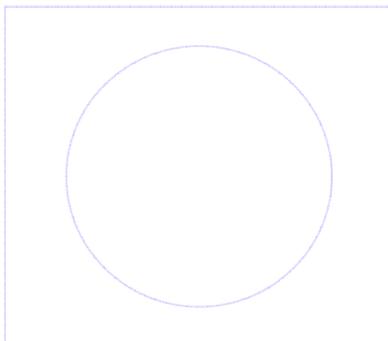


Image vectorielle

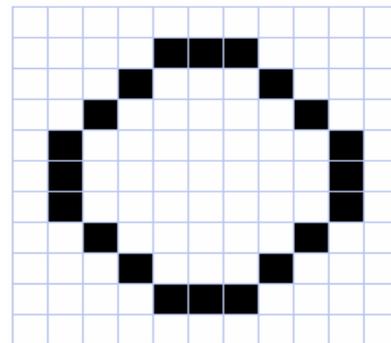


Image bitmap

7 Techniques de traitement d'images

Sur l'image, un certain nombre de manipulations sont possibles. Citons en quelques une :

- La restauration d'images.
- L'amélioration d'images.
- L'analyse descriptive des images.
- Le codage et la compression d'images.

Restauration d'images

Si l'image après compressée pour permettre une transmission ou un stockage plus aisés, il faut après réception la remettre en forme. Même en l'absence de compression certaines restaurations sont nécessaires pour corriger les erreurs dues à l'échantillonnage ou aux bruits. Il s'agira donc de s'approcher le plus possible de l'image originale.

On peut aussi vouloir améliorer l'image pour l'adapter soit à l'homme soit à l'ordinateur afin d'obtenir une image structurelle.

Amélioration d'images

L'amélioration d'images consiste en un ensemble de méthodes destinées à améliorer l'aspect visuel d'une image (évaluation de la qualité des scènes, corrections radiométriques et géométriques). Ces méthodes appelées pré-traitements et post-traitements regroupent un ensemble de techniques qui visent à atténuer les effets indésirables. En effet, les images ne peuvent pas échapper aux effets de dégradation dus essentiellement soient aux phénomènes physiques tels que - La diffraction du système optique, - Le flou dû au mouvement de l'image durant son acquisition.

Soient aux erreurs survenues lors du processus de codage. Les principaux traitements liés à l'amélioration d'images sont :

- La modification d'intensité des points de l'image,
- Le renforcement du contraste de l'image,
- La réduction des bruits contenus dans l'image.

Le filtrage est l'une des méthodes développées pour pallier à ces dégradations. Il permet d'éliminer les parasites, tel que le bruit et d'améliorer la qualité des images. Les méthodes de filtrage sont des opérations locales qui consistent à remplacer la valeur de chaque pixel de l'image par une combinaison linéaire des valeurs de ses voisins.

Analyse descriptive des images

L'analyse des contours, de la texture, l'analyse quantitative et morphologique des objets, correspondent à la phase d'extraction d'informations.

La segmentation de l'image est la séparation des divers éléments de celle-ci en régions connexes ayant mêmes propriétés, les régions peuvent être caractérisées par leurs frontières, c'est dans ce cas la segmentation par extraction de contours, ou bien être directement caractérisées par les pixels qui les composent, il s'agit alors de la segmentation en régions homogènes. Les propriétés constituant le critère de segmentation peuvent être choisies de manière à effectuer une segmentation en régions ayant même niveau de gris, même texture, même couleur, etc,...

Compression des données

La compression est un domaine qui connaît, depuis plusieurs années, une importance accrue, et qui conduit à une grande diversité d'approches et d'algorithmes. En informatique, la compression consiste à réduire le volume des données à stocker ou à transmettre en codant les données sous un format différent occupant moins d'espace que celles d'origine.

Objectifs de la compression des données

La compression de données touche deux principaux objectifs :

1. La réduction de l'espace de stockage des données,
2. Accroître le taux d'efficacité des lignes de transfert de données, ce qui permet de :
 - a) Augmenter la vitesse apparente de transmission.
 - b) Diminuer le coût financier du matériel.

REFERENCES

1. Léger. A, « La norme JPEG » Proceedings des journées CCETT sur la compression des signaux audiovisuels, Janvier 1992.
2. Chen. W.H and C.H. Smith, « Adaptive coding of monochrome and color images » IEEE Trans. On Communications, vol. 25, n°. 11, pp. 1285-1292, Nov 1977.
3. Chang. C.Y, wok. R. K and Curlander. J. C, « Spatial compression of Seasat SAR imagery » IEEE Trans. Geosc. and Remote Sensing , vol. 26, n°. 5, pp. 673-685, Sep 1988.
4. Andreadis. A, Benelli. G and Susini. S, « A DCT - based adaptive compression algorithm customized for radar imagery» IEEE (1997), pp. 1993.1995.
5. Marion. A, « Introduction aux techniques de traitement d'images » Editions Eyrolles, Paris, 1987.
6. Boite. R et M. Kunt, « Traitement de la parole »
7. Gamaz. N, « Etude et développement d'algorithmes de compression d'images fixes par TCD.2D » Thèse de Magister en Cybernétique, Nov 1992.
8. BENMERABET. A, « Analyses multirésolution et multiéchelle appliquées aux traitements d'images satellitaires » Thèse de Doctorat d'état en électronique. USTHB, 2001.
9. Guichard. J et Nase. D, « L'image numérique et le codage » L'écho des recherches, n°. 126, 4^{ème} trimestre, 1986.
10. Boudissa. E. G, « Fractales et vidéo : Implémentation d'un algorithme mixte de compression de séquences vidéos » Mémoire de Magister en automatique industrielle, Tizi-Ouzou .
11. Baskurt. A et Goutte. R, « Compression d'images numériques par la TCD » Thèse de Doctorat en Informatique INSA, Lyon (France) 1989.

12. Katto. J, Onda. K and Yasuda.Y, «Variable bitrate coding based on human visual system »
Signal processing : Image communication 3 (1991) 313-320.
13. Netravali. A.N and Haskell. B.C, « Digital pictures representation and compression »
Application of communications theory series edition : R.Wlucky, 1977.
14. Comes. S, Maes. S and Droogenbroeek. V, « Recent and Prospective decorrelation
technique for image processiong » Ann.Telecomm, vol 48, n°.7-8, 1993.
15. Nelson. M , « Compression de données, textes, images et sons » DUNOD, 1993.
16. Abdat. M, « Etudes des techniques de compression des images fixes et amélioration de la
résistance aux erreurs de transmission » Thèse de Doctorat, au conservatoire national
des arts et métiers, Dec 1995.
17. Puimatto. G , « Notions sur la compression des données dans les applications
multimédias » S.I.E, 1994.
18. Welch. T.A, « A technique for high performance data compression » IEEE Computer,
vol. 17, n°. 6, pp. 8-19, June 1984.
19. Jain. A.K, « Image data compression » Proceedings of the IEEE, vol. 69, n°. 3, March
1981.
20. Ahmed. N , Natarjan. T and Rao. K.R, « Discrete cosine transform » IEEE Transactions
on Computers Januray 1974, pp. 90-93.
21. Huang. J.Y and Schulthneiss. P.M, « Block quantization of correlated gaussian random
variables » IEEE Transaction, Comm. Systems, September 1963.
22. Nebout. C.L, Latry. C et Moury. G, « La compression d'images embarquées pour les
missions spatiales » ANN.Télécommun., vol. 56, n°. 11-12, pp. 632-645, 2001.
23. Mallat. S, « A theory for multiresolution signal decomposition : The Wavelet
representation » T-PAMI (11), 1989, pp. 674-693.
24. Coben. A, Daubechies. I and Feauveau. J.C, « Biorthogonal bases of compactly
supported Wavelets » Comm. Pur. Appl. Math., n°. 45, pp. 485-560, 1992.
25. Saphiro. J. M, « Embedded image coding using zerotrees of wavelet coeficients » IEEE,
Trans on signal processing, vol. 41, n°. 12, Dec 1993.

26. Said. A and Pearlman. W. A, « A new fast image codec based on set partitioning in hierarchical trees » IEEE, Transactions on image processing, vol. 5, n°. 9, pp. 1303-1310, Sept 1996.
27. Fiche. P et Ricordel. V, « Etude d'algorithmes de quantification arborescente pour la compression d'images fixes » IRISA, Publication Interne, n°. 87, 1994.
28. Gray. R.M, « Vector quantization » IEEE, ASSP magazine, pp. 4 -29, April 1984.
29. Linde. J, Buso. A and Gray. R.M, « An algorithm for vector quantizer design » IEEE Transactions Communications, vol. 28, n°. 1, pp. 84-95, Janvier 1980.
30. Ngono. J.M, « Compression des images de Radar à synthèse d'ouverture dans le cadre de leur utilisation dans les systèmes d'information géographique » Thèse de Doctorat en sciences de l'information géographique -Traitement du signal et télécommunications.
31. Wintz. P. A, « Transform picture coding » Proc. IEEE, vol. 60, n°. 7, pp. 809-820, July 1972.
32. Cruz. S, Grosbois. R et Ebrahimi. T, « JPEG 2000 – la nouvelle norme pour le codage d'images » Ecole polytechnique – Fédérale de Lausanne, Mar 2001.