

# Principales Normes de compression sans pertes d'images fixes.

Mme Amel Benazza-Benyahia  
Maître-assistante  
Département de Mathématiques Appliquées, Signal et Communications,  
SUP'COM,  
3.5 Km Route de Raoued, 2083 Cité El-Ghazala, Tunis, TUNISIE,  
Tel: (+216)1 856-440, Fax: (+216)1 856-829  
Email: [ben.yahia@planet.tn](mailto:ben.yahia@planet.tn)

Version de Mai 2000

# Table des matières

<b>1</b>	<b>Contexte de l'étude</b>	<b>6</b>
1.1	Compression sans ou avec pertes . . . . .	6
1.2	Images analogiques/numériques . . . . .	6
1.3	Avantages de la numérisation . . . . .	7
1.4	De la nécessité de comprimer . . . . .	8
1.5	De la nécessité de normaliser . . . . .	8
<b>2</b>	<b>Procédure de normalisation</b>	<b>10</b>
2.1	Instances de normalisation internationales . . . . .	10
2.2	Instances de normalisation nationales (USA) . . . . .	10
2.3	Phases de développement d'une norme . . . . .	11
<b>3</b>	<b>JPEG en mode sans pertes</b>	<b>12</b>
3.1	Instance de normalisation . . . . .	12
3.2	Attentes sur la norme . . . . .	12
3.3	Principe de la prédiction . . . . .	13
3.4	Les prédicteurs de la norme . . . . .	13
3.5	Codeur binaire . . . . .	15
3.6	JPEG évolue toujours! . . . . .	15
3.7	Performances . . . . .	15
3.8	Besoin d'une autre norme! . . . . .	15
<b>4</b>	<b>JPEG-LS</b>	<b>17</b>
4.1	Historique . . . . .	17
4.2	Principe de la norme . . . . .	17
4.3	Prédiction . . . . .	17
4.4	Détermination des contextes . . . . .	18
4.5	Modèle de la distribution des erreurs de prédiction . . . . .	18
4.6	Codes de Golomb . . . . .	19
4.7	Performances . . . . .	19
<b>5</b>	<b>JPEG2000</b>	<b>21</b>
5.1	Contexte de l'étude . . . . .	21
5.2	Historique . . . . .	21
5.3	Attentes à propos de la norme . . . . .	22
5.4	Schéma-bloc de la partie I . . . . .	23
5.4.1	Transformation en ondelettes réversible . . . . .	24
5.4.2	Codeur entropique . . . . .	26
5.5	Premiers résultats . . . . .	26
5.5.1	Versions préliminaires . . . . .	26

5.5.2	Complexité . . . . .	26
5.5.3	Amélioration des taux de compression . . . . .	26

# Liste des tableaux

3.1	Les sept prédicteurs de la norme JPEG en mode sans pertes. . . . .	14
4.1	Quelques exemples de codage d'un code de Golomb $G_2$ . . . . .	19
4.2	Débits moyens en bits/pixel (moyennes sur les plans de couleur) des deux normes, extrait de [8]. . . . .	20
5.1	Facteur moyen de réduction des tailles d'images compressées entre JPEG2000 et JPEG en mode sans pertes. . . . .	26

# Table des figures

1.1	Chaîne de compression sans pertes d'une image analogique $I_a$ . . . . .	7
3.1	Voisinage du pixel courant $(m, n)$ pour le calcul de la prédiction. . . . .	14
3.2	(1) Image originale "mri.8bit" de taille $256 \times 256$ , codée sur 8 bits/pixel, issue de la base de données de JPEG. (2) Histogramme de l'image originale "mri.8bit". . . . .	16
3.3	(1) Image erreur de prédiction de "mri.8bit" obtenue par application du prédicteur 6. L'image a subi un zoom d'un facteur 7. Le débit moyen pour la coder est de 3.4814 bits/pixel. (2) Histogramme de cette image erreur de prédiction. . . . .	16
5.1	Codeur du VM de la norme JPEG2000 (en mode sans pertes). . . . .	23
5.2	Décodeur du VM de la norme JPEG2000. . . . .	23
5.3	Du haut vers le bas de la gauche vers la droite : images $a_1, h_1, v_1$ et $d_1$ obtenues après décomposition en ondelettes 5/3 de l'image "mri.8bit". Les sous-images des détails ont été amplifiées d'un facteur 10. . . . .	25
5.4	Du haut vers le bas de la gauche vers la droite : histogrammes de images $a_1, h_1, v_1$ et $d_1$ obtenues après décomposition en ondelettes 5/3 de l'image "mri.8bit". Les sous-images des détails ont été amplifiées d'un facteur 10. . . . .	25

# Avant-propos

Ce document se propose de décrire succinctement les principales normes de compression sans pertes d'images fixes. Ces normes intéressent particulièrement la communauté des médecins pour le stockage et/ou la transmission des images acquises lors des examens radiologiques.

L'auteur de ce document n'appartient pas à la communauté des médecins. Toutes les remarques et suggestions des lecteurs sont les bienvenues car elles ne peuvent que contribuer à améliorer cet article. Elle a délibérément omis les descriptions trop techniques en veillant à souligner l'ampleur de l'activité des traiteurs d'images, des informaticiens dans le domaine de la compression d'images. Ce document ne se suffit pas à lui même. Il est destiné à initier les profanes et à convaincre les (rares !) sceptiques des potentialités qu'offre le "numérique" au monde médical en terme de gestion, transfert de données ... Il serait souhaitable de compléter la lecture de ce document par la consultation des sites spécialisés.

Comme la plupart des documents sont rédigés en langue anglaise, il a été jugé opportun de mentionner en italique la traduction de la plupart des termes techniques.

# Chapitre 1

## Contexte de l'étude

Ce document se propose de dresser un panorama des normes actuelles de compression (ou de codage) sans pertes d'images. Avant de décrire ces normes, il s'avère impératif de préciser quelques notions élémentaires relevant du domaine de la compression.

### 1.1 Compression sans ou avec pertes

Une méthode de compression est dite “sans pertes” ou conservatrice (en anglais, *lossless*) si elle garantit que l'image décompressée est l'exacte copie de l'image originale. Il est important de préciser que même si l'image décompressée est visuellement proche de l'image originale, il n'en reste pas moins qu'il faut vérifier l'égalité des amplitudes point par point. A l'opposé, une méthode de codage est dite “avec pertes” (*lossy*) si elle autorise une distorsion de l'image décompressée par rapport à l'image originale. La différence entre les méthodes de compression sans et avec pertes se situe au niveau des valeurs des taux de compression <sup>1</sup>. En effet, la contrainte de fidélité exacte de la reconstruction limite souvent les taux de compression en moyenne autour de 2. En revanche, le fait de tolérer des distorsions au niveau de l'image restituée permet d'atteindre des taux de compression bien plus élevés (parfois, supérieurs à 10). Le choix entre les deux types de méthodes dépend essentiellement de l'application considérée. Par exemple, dans des applications de visiophonie ou de vidéoconférence, des méthodes de codage avec pertes permettent de réduire le flot de données à transmettre qui sans cela, dépasserait largement les capacités des canaux de transmission habituels. Toutefois, dans des applications spécifiques comme l'imagerie médicale ou encore l'imagerie satellitaire de télédétection, il s'avère indispensable de préserver l'intégrité des données. En effet, l'apparition du moindre artefact risquerait de fausser le diagnostic du praticien ou l'interprétation des clichés satellitaires. Actuellement, dans de nombreux pays, aux termes de la loi, la compression des images médicale numériques à des fins de stockage doit garantir une reconstruction rigoureusement exacte. C'est pour cette raison que les efforts actuels en matière de compression d'images médicales concernent quasi exclusivement, les méthodes sans pertes.

### 1.2 Images analogiques/numériques

Cependant, malgré cette contrainte légale, la situation actuelle des principaux services d'imagerie médicale doit être prise en compte. En effet, dans de tels services, il est possible de distinguer deux types d'images selon leur mode d'acquisition. En effet, la première catégorie comprend les images dites analogiques (*analog*) dont les amplitudes prennent des valeurs continues. Les images classiques de radiologie à base de rayon X sont des exemples typiques d'images analogiques. La seconde classe d'images concerne les images qui ont été initialement acquises sous forme numérique.

---

<sup>1</sup>On appelle taux de compression le rapport entre la taille de l'image compressée et la taille de l'image originale.

Les images numériques (*digital*) se présentent sous forme de matrices de taille  $M \times N$ . Chaque élément à la ligne  $m$  (pour  $m = 1, \dots, M$ ) et à la colonne  $n$  (pour  $n = 1, \dots, N$ ) de la matrice est appelé pixel (pour *picture element*). Il est associé à un niveau d'intensité  $I(m, n)$  appartenant à un ensemble fini d'amplitudes. Ces niveaux d'intensité sont représentatifs des propriétés physiques et/ou chimiques de la structure des objets examinés. Par exemple, en tomographie à rayons X, les niveaux de gris correspondent aux coefficients relatifs d'atténuation linéaire des tissus considérés. Il est important de souligner qu'une image analogique peut être numérisée grâce au recours à des numériseurs. Ceux-ci peuvent être des scanners au laser, des caméras numériques, ... C'est le scanner à laser qui est le plus couramment utilisé car c'est le numériseur qui préserve le mieux la résolution des films analogiques. Même si les scanners présentent des résolutions de plus en plus élevées, il n'en reste pas moins que le processus de numérisation entraîne une perte irréversible d'information par rapport au cliché analogique initial même si cette dégradation n'est pas visuellement décelable. En effet, la conversion analogique/numérique met en œuvre une quantification (*quantization*) des niveaux d'amplitudes. Le rôle du quantificateur (*quantizer*) est de discrétiser les niveaux d'amplitude (typiquement sur 256 à 4096 niveaux). Or, la quantification est une opération non-inversible c.à.d. qu'il est impossible de restituer exactement le signal d'entrée à partir du signal quantifié. Ainsi, le quantificateur qui est un des organes du convertisseur analogique/numérique est la source d'une distorsion. Néanmoins, une fois la numérisation d'une image analogique quelconque  $I_a$  accomplie, on dispose d'une image numérique  $I_n$  en général, sous forme d'un fichier dans la mémoire du PC relié au scanner et toute méthode de compression de  $I_n$  doit se faire sans pertes. A partir de l'image compressée  $I_c$ , l'étape de restitution permet de générer l'image décompressée  $I_d$  telle que  $I_d = I_n$  comme le montre la figure 1.1.

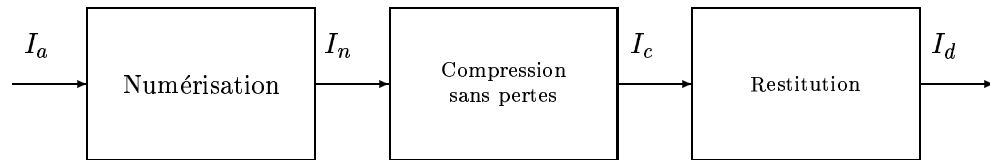


FIG. 1.1: Chaîne de compression sans pertes d'une image analogique  $I_a$ .

### 1.3 Avantages de la numérisation

Dans le domaine médical, la tendance actuelle est de généraliser les modes d'acquisition sous forme numérique et d'intégrer le processus de numérisation dans les capteurs du système d'acquisition. Il est important de souligner que cette évolution se justifie par les avantages indéniables que présentent les systèmes numériques de traitement d'images sur les systèmes analogiques classiques. Sans être exhaustif, citons les principaux atouts de la numérisation :

- la souplesse des traitements (zoom, augmentation de contraste, sélection des zones d'intérêt dans l'image, ... ),
- la facilité de dupliquer les clichés numériques, la mise en mémoire étant d'une fidélité parfaite, l'image peut être utilisée à un autre instant que celui où elle a été créée par le balayage,
- l'archivage fiable de ces clichés, s'affranchissant du problème de vieillissement des supports traditionnels d'archivage,
- une meilleure communication des données entre les praticiens grâce à un transfert aisé des images notamment à des fins de télé-expertise,

- le télé-enseignement pour la formation des promotions de futurs médecins,
- la mise en commun au niveau de la communauté des médecins, des images relevant par exemple, de cas cliniques saillants à travers la constitution de banques d’images spécialisées,
- la possibilité d’archiver le dossier d’un patient, de le mettre à jour, au besoin de le transmettre instantanément en cas d’urgence,
- l’exploitation efficace et aisée du contenu de ces banques d’images pour l’élaboration d’études statistiques de grande envergure.

## 1.4 De la nécessité de comprimer

Malgré tous ces avantages, les systèmes numériques de traitement d’images ont l’inconvénient de générer un flot de données important. Quelques chiffres suffisent à souligner l’ampleur du phénomène. Dans l’impossibilité d’évaluer la quantité de données numériques en Tunisie, nous limiterons au cas nord-américain [1]. En 1995, environ 70% des examens radiologiques aux USA se sont faits en mode analogique et ont concerné essentiellement la radiologie par rayons X. Les 30 % des examens restants étaient réalisés directement sous forme numérique (tomographie, résonance magnétique, mesures ultrasonores, médecine nucléaire, angiographie numérique soustractive ... ). La quantité d’information numérique générée annuellement aux USA est de l’ordre de  $10^{15}$  octets. Ainsi, un hôpital de 1500 lits peut produire jusqu’à 20 téraoctets/an. Ces données doivent être mémorisées sur des supports d’archivages à capacité maximale. Les disques optiques peuvent contenir jusqu’à 2 téraoctets mais ils ont l’inconvénient d’être coûteux. De même, la transmission d’une telle quantité d’information ne va pas sans poser des problèmes. En effet, il faut 13 secondes pour transmettre à travers un réseau à haut débit (20 Mbits/s), une radiologie des poumons de taille  $4000 \times 4000$  codée sur  $2^{12}$  bits/pixel. Or, des études ont montré que les praticiens ne tolèrent pas des délais d’affichage supérieurs à 2 secondes. Ainsi, s’avère-t-il indispensable de recourir à une opération de compression sans pertes. Celle-ci aura pour but de réduire la place mémoire requise pour l’archivage des images ou de diminuer les temps de transmission des clichés tout en préservant l’intégrité des données. Dans tout ce qui suit, sauf mention explicite, les méthodes de compression considérées sont des méthodes sans pertes.

## 1.5 De la nécessité de normaliser

Dès l’avènement des technologies numériques, plusieurs méthodes de compression sans pertes ont été développées. Elles n’ont pas toujours concerné les images médicales. Tout d’abord, il est utile de replacer dans leur contexte historique les premiers travaux sur la compression d’images. En effet, les besoins en codage sans pertes se sont fait sentir au niveau de la communauté des informaticiens dans les années 70. En effet, l’enjeu était de réduire le nombre de bits utilisés pour le stockage des fichiers texte (en binaire ou en ASCII). Ces travaux ont donné naissance aux célèbres algorithmes ZIP et COMPRESS (utilitaire de compression sous le système d’exploitation Unix). Ces algorithmes sont disponibles gratuitement <sup>2</sup>. Ces algorithmes sont conçus pour coder des fichiers texte plutôt que des images [2]. Ils exploitent les caractéristiques statistiques des données (notamment, la fréquence d’apparition des niveaux de gris) en tenant compte des résultats théoriques en matière de codage source relevant de **la théorie de l’information**. Cette dernière branche a pour objectif l’étude des codeurs à partir d’un modèle statistique des données considérées. Par ailleurs, à la fin des années 70, les normes de transmission existantes de facsimilés (images noir et blanc) concernaient uniquement le mode analogique. La norme du groupe G1 (1968) utilise pour la transmission, la modulation de fréquence analogique qui accepte une échelle de niveau de gris. En

---

<sup>2</sup>Pour Zip, on peut consulter le site <http://quest.jpl.nasa.gov/Info-Zip>

1976, le groupe G2 finalise la recommandation T3 qui reprend les grandes lignes du mode G1 mais le temps de transmission (analogique) est diminué de moitié grâce à une modulation d'amplitude. A cette époque, les instances de normalisation du Comité Consultatif International Télégraphique et Téléphonique (CCITT) ont reconnu les potentialités des technologies numériques émergentes pour la transmission **numérique** des facsimilés. Elles ont initié des travaux de normalisation qui ont abouti aux 2 normes suivantes :

- recommandation T4 en 1980 pour les télécopieurs du groupe 3 [3],
- recommandation T6 en en 1984 pour les télécopieurs du groupe 4 [4].

Ces normes satisfont la contrainte de faisabilité en temps réel. Un délai d'une minute pour le codage/décodage d'une image haute-résolution sur un canal à 4800 bit/s a été jugé satisfaisant par le CCITT. Cela correspondrait à un débit comprimé de 0.07 bpp. Rappelons que si aucune compression n'est réalisée, la transmission de ce document sur un tel canal durerait 14.3 minutes! Ces chiffres mettent bien en évidence l'utilité de la compression.

Dans ce contexte de compression d'images ayant un faible nombre de niveaux (typiquement de 2 à 6), le CCITT conjointement avec l'organisation internationale de normalisation (*International Standard Organization*, ISO) a produit les normes JBIG et sa version améliorée JBIG2 [5]. Il s'agit d'un acronyme issu de *Joint Bilevel Image Experts Group*. La norme JBIG permet un gain en terme de compression de l'ordre de 30% par rapport à la norme T6, pour les images binaires.

De manière générale, on peut dire que la compression est indispensable pour la transmission et/ou l'archivage des images. Comme les images présentent un contenu imprévisible, de nombreuses méthodes ont vu le jour. Cette diversité est inquiétante car elle risque de laisser place à des systèmes incompatibles. Il s'avère nécessaire de procéder à une normalisation des systèmes de codage d'images en place pour garantir l'inter-opérabilité des différents systèmes de télécommunications. Cette tâche est difficile car les paramètres (techniques et financiers) sont souvent contradictoires. Mais avant de continuer la présentation des normes actuelles de compression sans pertes, il semble opportun de décrire succinctement le processus même de normalisation.

## Chapitre 2

# Procédure de normalisation

### 2.1 Instances de normalisation internationales

Plusieurs instances internationales de normalisation existent. Celles qui intéressent la compression d'images sont les suivantes.

- La commission internationale d'électrotechnique (*International Electrotechnical Commission, IEC*) regroupe environ 41 pays, plus de 80 comités technique (*Technical Committees, TC*), plus de 100 sous-commissions (*Sub-Commission, SC*) et près de 700 groupes de travail (*Working Groups, WG*).
- L'organisation internationale de standardisation (*International Organization for Standardization, ISO*) réunit environ 75 pays, 150 TC., 600 SC et 1500 WG. Les membres de l'ISO s'organisent sous la forme de délégations nationales (*national bodies*). Les décisions sont faites par consensus parmi tous les groupes avec une approbation finale des chefs de délégations.
- L'union internationale des télécommunications (ex-CCITT, *International Telecommunication Union, ITU*) collabore avec l'ISO.
- Le comité technique conjoint sur la technologie d'information (*Joint Technical Committee on Information Technology, JTC1*) collabore avec l'ISO et l'IEC

### 2.2 Instances de normalisation nationales (USA)

En marge de ces instances de normalisation internationales, chaque pays a la possibilité de constituer et d'animer des entités de normalisation qui font appel aux compétences d'experts nationaux. Par exemple, en France, l'Association Française de NORmalisation (AFNOR) est une entité nationale se chargeant de toute sorte de normalisation en particulier en matière de compression d'images. Les USA se distinguent par le nombre et la puissance de ses instances de normalisation. Il semble judicieux de citer les plus importantes.

- L'institut national américain des standards (*American National Standards Institute, ANSI*) accrédite toutes les organisations de normalisation aux USA. Il regroupe 1300 compagnies, 270 membres associés, 25 membres gouvernementaux.
- Le comité national pour les standards de technologie d'information (ex-X3, *National Committee for Information Technology Standards, NTCITS*) est accrédité par l'ANSI. En effet, certains TC servent aux SC de JTC1 en tant que comités de conseils techniques (*Technical Advisory Committees, TAG*).

## 2.3 Phases de développement d'une norme

Il est possible de distinguer 3 étapes lors de l'élaboration d'une norme : l'étape de divergence, l'étape de convergence et enfin l'étape de vérification et d'optimisation.

L'étape de divergence se fixe les buts suivants : explorer les possibilités techniques et identifier les techniques prometteuses menées indépendamment par les laboratoires. A la fin, les grandes lignes de développement du standard sont mises en évidence.

L'étape de convergence vise l'élaboration d'une méthode de codage reconnue par tous. Il s'agit d'un travail de collaboration intense, les mêmes résultats doivent être obtenus sur des machines et des langages de programmation différents. A l'issue de cette étape, un modèle de codage initial (référence) est défini dans sa généralité. Des améliorations sont apportées (nouveaux éléments, optimisation de paramètres, . . . ) et on met à jour le modèle de référence lors de rencontres internationales périodiques. Cette procédure est itérée jusqu'à l'obtention de performances satisfaisantes. Enfin, l'étape de vérification/optimisation a pour objectif de confirmer les spécifications ambiguës et de comparer entre 2 ou 3 codecs *hardware* indépendants.

# Chapitre 3

## JPEG en mode sans pertes

### 3.1 Instance de normalisation

A la fin des années 80, le CCITT et l'ISO et le JTC1 ont obtenu le soutien des groupes industriels et des universitaires concernés pour définir une norme internationale pour la compression d'images fixes (*still images*). Le groupe d'étude qui a été mis en place en 1986, est le *Joint Photographic Expert Group* (JPEG). Plus précisément, il s'agit du groupe ISO/IEC/JTC1/SC29/WG10<sup>1</sup>. Ce groupe a proposé la norme IS 10918-1 (ou ITU-T T.81) qui porte son nom et qui a été publiée en 1994 [6], [7].

### 3.2 Attentes sur la norme

Dès le début, les experts ont défini une sorte de “cahier de charges” à satisfaire par la méthode de compression normalisée.

1. La qualité visuelle de l'image reconstituée est paramétrable. Les données sont d'autant plus compressées que la qualité attendue est faible. Toutefois, la qualité doit pouvoir atteindre la qualité photographique.
2. La norme doit être aussi proche que possible de l'état de l'art existant et assurer une compatibilité avec les autres systèmes.
3. Aucun format d'images n'est imposé. La compression doit être applicable sur toute image numérique quelconque. Ainsi, les spécifications n'indiquent ni la taille de l'image, ni le système de représentation des couleurs. Elle propose de définir des plans-mémoires associés aux composantes de la couleur.
4. La méthode doit être applicable quel que soit le type d'image et indépendante de l'application considérée. Ainsi, elle ne doit utiliser aucune hypothèse sur le type d'image, aucun modèle *a priori* d'image. Les applications les plus diverses sont visées : applications multimédias, photovidéotexte, applications graphiques, cartographie, imagerie médicale, ...
5. La méthode doit être de complexité opérationnelle. Le coût opératoire doit être équilibré entre le codeur et le décodeur.
6. Selon leur convenance, les utilisateurs ont le choix entre une implémentation matérielle ou logicielle. L'implémentation logicielle doit être possible sur une large gamme de processeurs. L'implémentation matérielle doit être de faible coût et, si possible, fonctionner en temps réel.
7. La méthode doit comprendre les 4 modes opérationnels suivants.

---

<sup>1</sup>Le site officiel est <http://www.jpeg.org>.

- Construction séquentielle (*Sequential build-up*). L'image est reconstruite ligne par ligne, de gauche vers la droite et du haut vers le bas et ce, à pleine résolution.
- Construction Progressive (*Progressive build-up*). L'image est entièrement reconstruite à résolution réduite puis raffinée progressivement. Plusieurs balayages (*multiple scans*) sont prévus pour permettre d'avoir des images de plus en plus détaillées. Les applications concernées par ce mode sont par exemple, les applications de consultation d'images à travers une banque d'images.
- Codage hiérarchique (*Hierarchical encoding*). Il s'agit du même principe que le mode progressif. Ce mode est utilisé dans un environnement ayant des équipements de différentes résolutions.
- Compression sans pertes (*Lossless Compression*).

Les trois premiers modes concernent le codage non-conservateur et la norme est célèbre pour son mode séquentiel avec pertes. Mais en imagerie médicale, notre attention porte sur le dernier mode.

### 3.3 Principe de la prédiction

La compression est effectuée grâce à l'exploitation des redondances que présentent les images. En effet, les zones homogènes de l'image correspondent à des pixels ayant des niveaux d'intensité semblables. On dit que les pixels sont corrélés. Dans de telles zones, l'intensité du pixel courant  $I(m, n)$  peut être estimée à partir d'une combinaison linéaire  $\hat{I}(m, n)$  des intensités des pixels voisins. Dans ce cas, on parle de **prédiction linéaire**. L'erreur de prédiction est la différence  $e(m, n) = I(m, n) - \hat{I}(m, n)$ , elle est d'autant plus proche de 0 que le voisinage du pixel  $(m, n)$  est homogène. Si les niveaux sont proches, la différence  $e(m, n)$  va être proche de 0 et aura une plage de variations plus étroite que celle des amplitudes absolues  $I(m, n)$ . Un plus petit nombre de bits suffit à la coder. C'est à ce niveau qu'a lieu la réduction de débit. Dès lors, il semble judicieux de transmettre non plus la valeur du pixel mais la différence de valeur calculée à partir des pixels voisins. Le décodeur reçoit l'erreur de prédiction, l'ajoute à la valeur  $\hat{I}(m, n)$  calculée par son propre prédicteur et restitue la vraie valeur de l'intensité. Il s'agit d'un schéma de codage prédictif différentiel. En termes plus spécialisés, cette méthode est appelée **Modulation à Impulsions et Codage Différentiel** (MICD). En anglais, on parle de *Differential Predictive Coding Modulation* (DPCM).

### 3.4 Les prédicteurs de la norme

La norme spécifie quels sont les pixels environnants le pixel courant  $(m, n)$  pour calculer la prédiction. Il s'agit du voisinage  $(a, b, c)$  comme le montre la figure 3.1. Par ailleurs, la norme spécifie 7 prédicteurs possibles comme le fait apparaître le tableau 3.1. Pour que la restitution soit sans pertes, la valeur prédite est une valeur arrondie grâce à l'opérateur  $\lfloor \cdot \rfloor$ .

Prédicteur	$\hat{I}(m, n)$
1	$a$
2	$b$
3	$c$
4	$\lfloor a + b - c \rfloor$
5	$\lfloor a - (b - c)/2 \rfloor$
6	$\lfloor b - (a - c)/2 \rfloor$
7	$\lfloor (a + b)/2 \rfloor$

TAB. 3.1: Les sept prédicteurs de la norme JPEG en mode sans pertes.

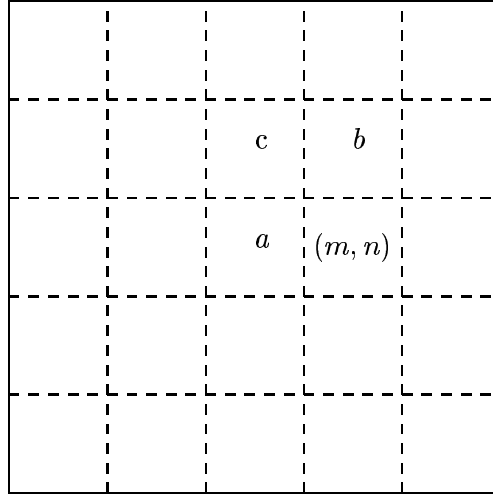


FIG. 3.1: Voisinage du pixel courant  $(m, n)$  pour le calcul de la prédiction.

Les équations permettant de calculer les valeurs prédites sont :

$$\left\{ \begin{array}{ll}
 \hat{I}(m, n) = I(m, n - 1) & \text{prédicteur 1} \\
 \hat{I}(m, n) = I(m - 1, n) & \text{prédicteur 2} \\
 \hat{I}(m, n) = I(m - 1, n - 1) & \text{prédicteur 3} \\
 \hat{I}(m, n) = I(m, n - 1) + I(m - 1, n) - I(m - 1, n - 1) & \text{prédicteur 4} \\
 \hat{I}(m, n) = I(m, n - 1) - \lfloor \frac{I(m - 1, n) - I(m - 1, n - 1)}{2} \rfloor & \text{prédicteur 5} \\
 \hat{I}(m, n) = I(m - 1, n) - \lfloor \frac{I(m, n - 1) - I(m - 1, n - 1)}{2} \rfloor & \text{prédicteur 6} \\
 \hat{I}(m, n) = \lfloor \frac{I(m, n - 1) + I(m - 1, n)}{2} \rfloor & \text{prédicteur 7}
 \end{array} \right. \quad (3.1)$$

*A priori*, un seul type de prédicteur est choisi pour toute l'image. Il s'agit du prédicteur qui garantit le meilleur de taux de compression qui est sélectionné.

## 3.5 Codeur binaire

Une fois l'erreur de prédiction obtenue, il faut veiller à la transcrire en binaire grâce à un codeur. Un codeur binaire a pour rôle d'affecter l'image  $e(m, n)$  une séquence de bits la plus petite possible afin de réduire la place mémoire pour son stockage et/ou diminuer le débit sur le canal de transmission. La norme JPEG laisse le choix entre deux codeurs connus que sont le codeur de Huffman et le codeur arithmétique. Par exemple, le codeur de Huffman procède à une conversion judicieuse des niveaux  $e(m, n)$  en mots de code binaire (*codewords*). La concaténation de ces mots constitue le flot binaire en sortie de la chaîne de compression. Ce flot binaire est concis car les niveaux  $e(m, n)$  les plus fréquents sont associés à des mots de code très courts. *A contrario*, on peut tolérer des mots de code longs pour les niveaux  $e(m, n)$  qui apparaissent rarement. Pour simplifier notre présentation, nous ne décrivons pas ces codeurs car il nous semble inutile de rentrer dans les détails "techniques" de ces codeurs d'autant plus que des exécutable sont disponibles gratuitement sur le réseau Internet.

## 3.6 JPEG évolue toujours !

Depuis 1996, la norme a fait l'objet d'extensions et d'améliorations de la norme concernant les modes de compression avec pertes. Un format d'images baptisé *Still Picture Interchange File Format* (SPIFF) a été défini pour faciliter les échanges d'images. Sur le site officiel du groupe JPEG, il est possible de télécharger le descriptif complet de ce format (annexe F).

## 3.7 Performances

Cette méthode de codage s'applique à toute image codée sur 2 à 16 bits/pixel. Néanmoins, ses performances dépendent du contenu de l'image originale. En effet, si l'image contient de grandes zones homogènes (typiquement des images à rayons X des mains), l'histogramme de l'erreur de prédiction  $e(m, n)$  présente un pic prononcé autour de 0. Une bonne compression est possible. En revanche, les performances sont moindres dans le cas des images détaillées, comportant de nombreux objets contrastés. A titre d'illustration, nous considérons l'image d'IRM "mri.8bit" de taille  $256 \times 256$ , issue de la banque d'images de JPEG (Fig 3.2-1). Ses pixels sont codés sur 8 bits : les niveaux de gris varient entre 0 et 255. On remarque que l'histogramme de l'image est multimodal (Fig 3.2-2). Un pic élevé en 0 est dû essentiellement à la contribution des pixels de l'arrière-fond qui est une zone homogène. Par application du jeu des 7 prédicteurs, il apparaît que le prédicteur 6 est celui qui fournit le meilleur taux de compression. L'image erreur de prédiction contient de larges zones noires : les intensités plus claires apparaissent au niveau des frontières des objets (Fig 3.3-1). Ce résultat était prévisible puisqu'au niveau des contours des objets, l'intensité lumineuse varie brusquement et les pixels voisins ne sont plus corrélés. La valeur de l'erreur de prédiction augmente par rapport à celle correspondant à une zone homogène. L'histogramme de l'image erreur de prédiction est monomodal, centré en 0 et présentant un pic élevé pour le niveau 0 (Fig 3.2-2). Après application d'un codage de Huffman, les intensités de l'image erreur sont codées en moyenne sur 3.4814 bits. Il en résulte un taux de compression d'environ 2.3 (c.à.d.  $8/3.4814$ ). Lors de tests de la norme, les taux de compression rapportés se situent en moyenne autour de 2.

## 3.8 Besoin d'une autre norme !

Les performances de la norme JPEG en mode sans pertes se sont révélées décevantes contrairement aux 3 autres modes avec pertes. Actuellement, la norme JPEG est utilisée surtout pour le mode séquentiel (avec pertes!) qui donne d'excellents résultats.

Avec l'avènement du réseau Internet et les besoins de stockage/transmission sans pertes d'images, les experts de JPEG se sont vus dans l'obligation de pallier la limitation de performances de la norme JPEG en mode sans pertes. Un appel à contribution a été lancé pour initier des travaux de normalisation concernant uniquement le mode sans pertes ou presque sans pertes (*near lossless*). Cette norme a été appelée la norme JPEG-Lossless (en abrégé JPEG-LS). Le chapitre suivant est consacré à sa description.

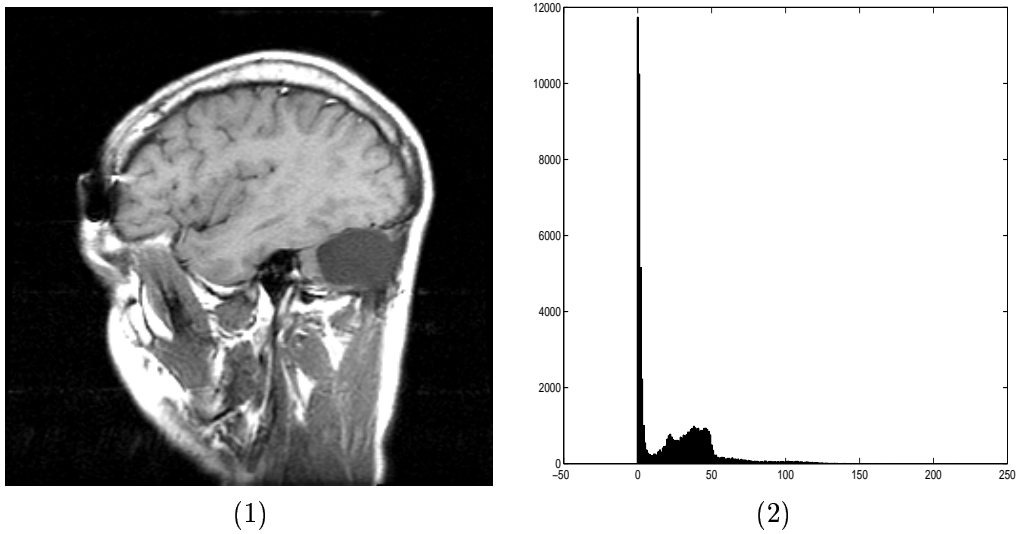


FIG. 3.2: (1) Image originale “mri.8bit” de taille  $256 \times 256$ , codée sur 8 bits/pixel, issue de la base de données de JPEG. (2) Histogramme de l’image originale “mri.8bit”.

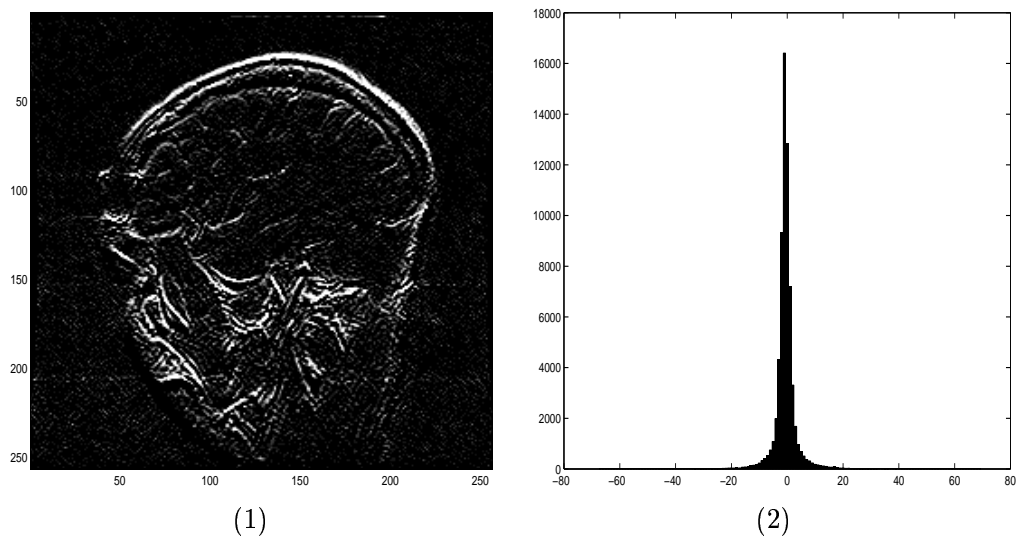


FIG. 3.3: (1) Image erreur de prédiction de “mri.8bit” obtenue par application du prédicteur 6. L’image a subi un zoom d’un facteur 7. Le débit moyen pour la coder est de 3.4814 bits/pixel. (2) Histogramme de cette image erreur de prédiction.

# Chapitre 4

## JPEG-LS

### 4.1 Historique

Le besoin de combler les déficiences de la norme JPEG en mode sans pertes ont conduit à l'adoption d'une nouvelle norme la norme JPEG-Lossless ou encore JPEG-LS. Cette norme émane de la méthode *LOw COmplexity LOssless COmpression for Images* (LOCO-I), proposée par des chercheurs de Hewlett-Packard [8]. Les exécutable des algorithmes de compression/décompression sont disponibles gratuitement à l'adresse suivante : <http://www.hpl.hp.com/loco>. Testée sur une grande variété d'images de différents types, la méthode LOCO-I se caractérise principalement par le meilleur compromis compression/complexité. Ainsi, elle dépasse en terme de compression JPEG en mode sans pertes (avec un codeur de Huffman) pour des niveaux de complexité comparable. De plus, elle est supérieure au codeur JPEG (avec un codeur arithmétique) en terme de complexité et de compression.

### 4.2 Principe de la norme

Cette méthode permet également le codage presque sans pertes (*near lossless*). C'est à l'utilisateur de définir un seuil de tolérance  $\eta$  en dessous duquel doit se maintenir l'erreur de codage. La méthode utilise un codage prédictif plus sophistiqué que celui mis en œuvre dans la précédente norme. Il s'agit d'un codage par contextes : le code binaire de l'erreur de prédiction dépend des pixels environnants ou encore du contexte du pixel considéré. De plus, ce codage par contextes utilise des modèles simplifiés de contextes afin de ne pas alourdir la charge calculatoire. A cet égard, les trois étapes suivantes sont envisagées.

- Une étape de prédiction du pixel courant  $I(m, n)$  utilise une fenêtre causale (*causal template*) c.à.d. un voisinage contenant des pixels déjà codés.
- La détermination d'un contexte dans lequel apparaît  $I(m, n)$ , ce contexte est une fonction de la fenêtre considérée.
- L'assignation d'un modèle pour la distribution des erreurs de prédiction conditionnellement au contexte de  $I(m, n)$ .

### 4.3 Prédiction

La valeur prédite est la contribution d'une composante fixe et d'une composante adaptative. La composante fixe  $\hat{I}_{med}(m, n)$  est définie par :

$$\hat{I}_{med}(m, n) \triangleq \begin{cases} \min(a, b) & \text{si } c \geq \max(a, b) \\ \max(a, b) & \text{si } c \leq \min(a, b) \\ a + b - c & \text{sinon} \end{cases}, \quad (4.1)$$

où on a posé :

$$a \triangleq I(m, n - 1) \quad b \triangleq I(m - 1, n) \quad c \triangleq I(m - 1, n - 1). \quad (4.2)$$

Les trois voisins du pixel  $(m, n)$  sont les mêmes que ceux utilisés par la norme JPEG en mode sans pertes. Ils sont représentés dans la figure 3.1. Ainsi, la valeur prédite  $\hat{I}_{med}(m, n)$  apparaît comme la valeur médiane des trois valeurs prédites  $I(m, n - 1)$ ,  $I(m - 1, n)$  et  $I(m, n - 1) + I(m - 1, n) - I(m - 1, n - 1)$ . Pour cela, on l'appelle détecteur de contour médian (*median edge detector*).

La seconde composante calculée adaptativement est un entier correspond à un prédicteur adaptatif affine. En fait, le calcul de cette composante est lié au choix du contexte.

## 4.4 Détermination des contextes

Le contexte qui conditionne le codage de l'erreur de prédiction courante est construit à partir des différences  $g_1$ ,  $g_2$  et  $g_3$  définies par :

$$d \triangleq I(m-, n + 1) \quad g_1 \triangleq d - b \quad g_2 \triangleq b - c \quad g_3 \triangleq c - a. \quad (4.3)$$

Ces trois différences permettent d'évaluer les variations d'intensité dans le voisinage du pixel courant. Elles permettent de connaître si le pixel courant est dans une zone homogène ou plutôt sur le contour d'un objet ce qui permet de contrôler le comportement statistique de l'erreur de prédiction. Pour réduire la charge de calculs, il est judicieux de quantifier les valeurs  $g_1$ ,  $g_2$  et  $g_3$  en un nombre peu élevé de niveaux  $-T, \dots, -1, 0, 1, \dots, T$ . Il y a donc  $(2T + 1)^3$  configurations possibles de contextes. Par ailleurs, en exploitant la symétrie des erreurs de prédiction, il est possible d'apparier les contextes de "signes opposés" : on diminue de moitié le nombre de contextes en le faisant passer à  $(2T + 1)^3/2$ . Pour la norme JPEG-LS, il a été choisi  $T = 4$  ce qui conduit à 365 contextes.

## 4.5 Modèle de la distribution des erreurs de prédiction

Il s'agit de déterminer la loi de probabilité de l'erreur de prédiction globale (fixe et adaptative) pour chaque contexte et le nombre total de contextes. Le nombre de paramètres du modèle permet de contrôler la complexité algorithmique. Le modèle de loi choisi est celui d'une loi géométrique :

$$P_{(\theta, \mu)}(\varepsilon) = (\theta^{1-\rho} + \theta^\rho)(1 - \theta)\theta^{|\varepsilon+\rho|}, \quad \varepsilon = 0, \pm 1, \pm 2, \dots, \quad (4.4)$$

où

- le paramètre  $\theta$  compris dans  $]0, 1[$  contrôle le taux de décroissance exponentielle de la loi,
- $\rho$  est un réel appartenant à  $[0, 1[$ .

## 4.6 Codes de Golomb

Il s'agit de transcrire en binaire les erreurs de prédiction qui suivent une loi géométrique dont les paramètres dépendent du contexte. Les codes de Golomb [10] ont été retenus car ils permettent le calcul des mots de code binaires sans recourir à des tables de conversion. En effet, cette famille de codes s'est avérée optimale pour coder les entiers positifs  $y$  suivant une loi géométrique de type  $(1-\theta)\theta^y$ . Or, l'équation 4.4 indique que la loi de l'erreur de prédiction globale a bien la forme d'une loi géométrique. Mais, avant de préciser leurs modalités d'application, décrivons succinctement les codes de Golomb.

**Définition :** Soit  $m$  un entier positif. On appelle code de Golomb  $G_m$  d'ordre  $m$  le code qui assigne à tout entier  $x \geq 0$  les deux parties suivantes.

- Les  $m$  bits les moins significatifs de  $x$  sont laissés tels quels.
- Les bits restants représentent un entier  $k$ . On les code en plaçant  $k$  "0" suivis d'un seul "1".

Le tableau 4.1 fournit quelques exemples de codage pour  $m = 2$ .

Entier $x$	Représentation binaire	$k$	Mot de code
19	$\underbrace{100}_{k} 11$	4	0000111
9	$\underbrace{10}_{k} 01$	2	00101
2	$\underbrace{-}_{k} 10$	-	110
0	$\underbrace{-}_{k} 0$	-	100

TAB. 4.1: Quelques exemples de codage d'un code de Golomb  $G_2$ .

**Application des codes de Golomb :** Un code de Golomb est utilisé pour chaque classe de la partition de l'espace  $(\rho, \theta)$ . On suppose que  $\rho \leq \frac{1}{2}$  puisque par suite de la transformation  $\varepsilon \rightarrow -(\varepsilon + 1)$  on peut se ramener pour les  $\rho > \frac{1}{2}$  à des valeurs de  $\rho < \frac{1}{2}$ .

Les entiers positifs sont entrelacés pour former la séquence  $0, -1, +1, -2, +2, \dots$ . Un code  $G_m$  est ensuite appliqué avec  $m = 2^c$ , où  $c$  désigne la classe du contexte associée à l'erreur de prédiction que l'on doit coder.

**Mode de codage par plages :** le mode de codage régulier (*regular mode*) décrit précédemment a une limitation quand il traite les zones homogènes et peut même conduire à un code excessivement long. Pour pallier cet inconvénient, un autre mode de codage a été prévu. Il s'agit du mode de codage par plages (*run mode*). Ce mode est retenu quand le pixel courant a la même (ou presque la même) amplitude que ses voisins pris dans un voisinage causal donné. Il est associé aux régions homogènes pour lesquelles la région de quantification de  $g_1, g_2$  et  $g_3$  est  $\{0\}$ . La sélection de ce mode est assurée en vérifiant que les valeurs quantifiées de  $g_1, g_2, g_3$  sont nulles. Une séquence du symbole  $a$  est alors trouvée et c'est sa longueur (pouvant être nulle) qui est codée par des codes de Golomb. Ce mode peut être interrompu par une fin de ligne ou encore par la "rencontre" d'un échantillon  $x$  différent de  $a$ . Dans ce cas, pour marquer l'interruption, on code la différence  $x - a$ .

## 4.7 Performances

L'équipe des chercheurs de Hewlett-Packard a mené des tests comparatifs entre les deux normes de compression sans pertes d'images [8]. Ces tests ont été effectués sur un grand nombre d'images ayant des contenus différents. Pour le mode sans pertes, il existe deux versions selon le choix du

codeur entropique. En effet, il est possible de considérer un codeur de Huffman ou un codeur arithmétique. Le tableau 4.2 montre que la norme JPEG-LS surpasse les deux versions basiques de la norme JPEG sans pertes. En moyenne, la norme JPEG-LS génère un débit inférieur de 22% au débit associé à la norme JPEG sans pertes (avec un codeur de Huffman).

Image	JPEG sans pertes codeur de Huffman	JPEG sans pertes codeur arithmétique	JPEG-LS
bike	4.34	3.92	3.63
cafe	5.74	5.35	4.83
woman	4.86	4.47	4.20
tools	5.71	5.47	5.08
bike3	5.18	4.78	4.38
cats	3.73	2.74	2.61
water	2.63	1.87	1.81
finger	5.95	5.85	5.66
us	3.77	2.52	2.63
chart	2.41	1.45	1.32
charts	4.06	3.07	2.77
compound1	2.75	1.50	1.27
compound2	2.71	1.54	1.33
aerial2	5.13	4.14	4.11
faxballs	1.73	0.84	0.90
gold	4.33	4.13	3.91
hotel	4.39	4.15	3.80
Moyenne	4.08	3.40	3.19

TAB. 4.2: Débits moyens en bits/pixel (moyennes sur les plans de couleur) des deux normes, extrait de [8].

# Chapitre 5

## JPEG2000

### 5.1 Contexte de l'étude

### 5.2 Historique

Le succès de la norme JPEG en mode avec perte ne doit pas faire oublier les lacunes de la norme en mode sans pertes. D'ailleurs, la norme JPEG-LS a été adoptée pour pallier les lacunes du mode de compression sans pertes. Là encore, les performances n'ont pas suivi les nouvelles attentes qui ont fait jour avec l'avènement de l'Internet sur lequel les échanges d'images prennent une part non négligeable.

Le projet d'une nouvelle norme a été adopté en 1996. Un appel à contribution public a suivi en mars 1997. En novembre 1997, le groupe de travail WG1 a augmenté de taille passant de 40 à 100 experts traduisant l'engouement des chercheurs et des industriels pour ce nouveau projet. Au commencement, 22 algorithmes-candidats ont été présentés (phase de divergence). Ils ont été systématiquement testés. Sur le plan quantitatif, ont été effectués des tests sur 22 images soigneusement sélectionnées et à différents débits (2,2,0.5,0.25,0.125 et 0.0625 bits/pixel). Sur le plan qualitatif, des tests subjectifs aux débits les plus faibles (0.25,0.125 et 0.0625 bits/pixel) ont été menés par la société Eastman Kodak avec la collaboration de 40 évaluateurs. A la suite de ces intenses simulations, il a été retenu l'algorithme dont les performances sont les meilleures pour toutes les configurations. Cela a donné naissance à un premier "canevas" de la norme que l'on désigne par modèle de vérification (*Verification Model*, VM). Depuis, à chaque session plénière, le groupe d'experts procède à des améliorations adoptées à la suite d'un consensus parmi les groupes et des approbations des chefs de délégations nationales.

La norme JPEG2000 est constituée des 5 parties suivantes.

1. Partie I : le "noyau" du système de compression (*Core coding system*). Aucune royauté n'est prévue car on estime qu'elle servira environ 80 % des applications JPEG2000. La publication de la norme se fera en décembre 2000.
2. Partie II : les extensions du système de compression. Il couvrira les 20 % d'applications restantes et des royalties devraient être perçues.
3. Partie III : *Motion* JPEG2000. Il s'appuiera sur un sous-ensemble d'outils définis dans la partie I pour prévoir le codage des images animées. Actuellement, les "profils" ou les dominantes de cette partie sont en cours de définition.
4. Partie IV : Test de conformité (*Compliance Testing*).
5. Partie V : *Reference Software*. Son but premier est de fournir une implémentation logicielle complète de la partie I. Il y a déjà deux propositions : Jasper et JJ2000. JJ2000 est le fruit de la collaboration entre Ericsson, le centre de recherche de Canon en France et l'institut

fédéral suisse de technologie. Il s'agit d'une implémentation sous Java du codeur/décodeur de JPEG2000.

Les travaux sur les parties I et II ont commencé en mars 1997 tandis que ceux relatifs aux parties restantes datent de décembre 1999. L'ensemble de ces études devraient s'achever en novembre 2001.

### 5.3 Attentes à propos de la norme

L'objectif prioritaire est d'atteindre des débits très faibles (inférieurs à 0.25 bits/pixel) même pour les images comportant beaucoup de détails. Les efforts ont été réalisés afin que la norme soit la plus flexible possible et qu'elle couvre le maximum de cas de figures d'applications.

1. La norme se veut une solution aux besoins actuels et vise un grand nombre d'applications telles que :
  - l'Internet,
  - le facsimilé,
  - l'impression,
  - la numérisation (*scanning*) tout-public ou en pré-presse,
  - la photographie numérique,
  - la télédétection (ou imagerie satellitaire),
  - les images sur les mobiles,
  - l'imagerie médicale,
  - les banques d'images numériques,
  - le commerce électronique.
2. La norme doit traiter aussi bien les images à plusieurs niveaux de gris et les images binaires. L'idée est d'utiliser les mêmes ressources d'un système donné pour prendre en considération les deux types d'images. On évite ainsi de recourir à deux normes comme par le passé avec JPEG et JBIG.
3. La compression doit pouvoir être avec ou sans pertes. Le même code binaire doit fournir de bonnes performances à la fois en mode avec pertes et en mode sans pertes.
4. Il est prévu une seule architecture du décodeur afin de faciliter les échanges entre différentes applications.
5. La nouvelle norme doit prévoir de nouvelles fonctionnalités qui ne sont pas offertes par les précédentes normes. Ces fonctionnalités sont les conséquences de la généralisation actuelle des outils de communication multimédia. Bien entendu, elles dépendent de l'application considérée.
  - Pour la consultation de banques d'images, la norme JPEG2000 doit permettre la reconstruction progressive des images. Elle consiste à décompresser des versions grossières de l'image originale. Si l'utilisateur le désire, un raffinement est effectué grâce à l'envoi d'un supplément d'information. Si au contraire, l'image affichée ne convient pas à l'utilisateur, celui-ci peut cesser sa consultation. Il aura accompli une économie en terme de délais de transmission car une image de moindre résolution est transmise plus vite qu'une image à résolution supérieure. Toutefois, cette reconstruction graduelle de l'image doit pouvoir aboutir à une image finale décompressée égale en tout point à l'image originale. Cette propriété de restitution progressive constitue une innovation par rapport aux précédentes normes qui se contentaient de décompresser l'image en une seule passe.

- La description d’une image doit se faire selon son contenu sans recours à des annotations textuelles. En effet, il s’avère difficile voire impossible de décrire une image par des mots-clés. Il faut donc développer des algorithmes permettant de calculer les caractéristiques pertinentes des images (texture, histogramme, ... ). Par la suite, ces caractéristiques serviront à accélérer la recherche à travers toute banque d’images. Par exemple, un docteur peut faire la requête d’extraire de la base de données les images contenant un certain type de tumeur. Il suffit alors de sélectionner les images ayant des caractéristiques semblables à celles de la requête. **L’indexation d’images** est la discipline qui se propose de développer des méthodes simples et rapides d’accès aux images.
- En imagerie médicale, l’accent doit être mis sur le mode sans pertes. La résolution élevée des images à coder doit être prise en compte (jusqu’à 16 bits/pixel). La norme doit offrir la possibilité de sélectionner dans l’image, les régions d’intérêt (*regions of interest*).
- La transmission des images sur les mobiles doit être robuste aux erreurs de transmission (milieu bruité, présence d’échos, ... ).
- En imagerie satellitaire, la norme doit être apte à coder des images de grande taille (supérieure à  $64\,000 \times 64\,000$ ). En général, pour une scène, il existe plusieurs clichés (les composantes spectrales) pris pas différents capteurs fonctionnant à des longueurs d’ondes adjacentes. On parle d’images hyperspectrales (si ce nombre dépasse la dizaine) et multispectrales, dans le cas contraire. La norme JPEG2000 doit pouvoir traiter des images ayant 500 composantes hyperspectrales ayant une précision de 8 jusqu’à 20 bits/composante.
- En numérisation, les caméras numériques doivent fonctionner en temps réel en matière de codage et les besoins en mémoire doivent être les plus faibles possibles.

## 5.4 Schéma-bloc de la partie I

Les figures 5.1 et 5.2 fournissent respectivement les schémas-bloc du codeur (sans pertes) et du décodeur constituant le VM. En mode avec pertes, il faut insérer un module de quantification avant l’étage de codage entropique.

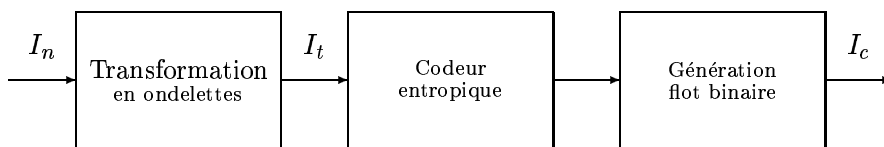


FIG. 5.1: Codeur du VM de la norme JPEG2000 (en mode sans pertes).

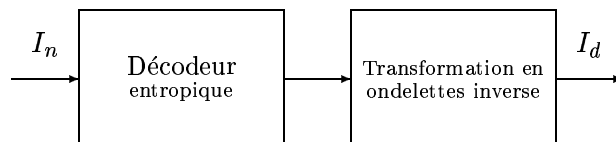


FIG. 5.2: Décodeur du VM de la norme JPEG2000.

Pour bien comprendre le fonctionnement de ce codeur-décodeur, il s'avère utile de passer en revue les différents modules qui le constituent.

### 5.4.1 Transformation en ondelettes réversible

A des fins de reconstruction progressive, il est impératif d'utiliser une représentation hiérarchique et multirésolution de l'image originale  $I_n$ . Plus précisément, une telle représentation consistera en :

- une sous-image constituant une version approximative (à résolution grossière) de l'image  $I_n$ ,
- plusieurs sous-images résiduelles à résolution croissante.

En effet, l'utilisateur commencera par afficher l'approximation puis s'il le désire, il procédera à un raffinement par l'affichage d'une version plus complète dont les détails ont été enrichis grâce à la réception de sous-images résiduelles. Ce processus est itéré un nombre de fois  $J$ , jusqu'à éventuellement, la reconstruction parfaite de l'image. La contrainte de reconstruction exacte équivaut à l'inversibilité de la transformation qui génère la représentation hiérarchique à partir de l'image originale  $I_n$ . De plus, il faut veiller que la reconstruction progressive conduit à des taux de compression plus élevés que ceux issus d'une compression directe (en une seule passe) de  $I_n$ .

Des études théoriques ont montré qu'une transformation qui conduit à une représentation multirésolution compacte est la transformation en ondelettes 5/3 dite de seconde génération. A ce niveau, afin de ne pas alourdir notre propos, nous jugeons inutile de fournir le détail des équations de la transformation. Concrètement, l'image  $I_n$  de taille  $M \times N$  est transformée en 4 sous-images  $a_1, v_1, h_1$  et  $d_1$  de même taille  $\frac{M}{4} \times \frac{N}{4}$ .

- La sous-image  $a_1$  est une version lissée de l'image  $I_n$ , ayant des contours plus atténués.
- Les sous-images  $v_1, h_1$  et  $d_1$  correspondent respectivement aux contours verticaux, horizontaux et diagonaux de  $I_n$ .

La figure 5.3 montre les 4 sous-images en question. On remarque bien que  $a_1$  est une version simplifiée de  $I_n$  et de moindre taille. Les trois sous-images résiduelles ont des niveaux élevés sur les contours et transitions des objets. On dit qu'elles représentent les composantes hautes-fréquences de  $I_n$ . La compression est possible en prenant en compte que la puissance du signal est concentrée sur la sous-image  $a_1$ . Dans le cas de l'image "mri.8bit",  $a_1$  contient 30 fois plus de puissance que les 3 autres sous-images de contours. On peut donc coder efficacement ces images résiduelles et économiser substantiellement le nombre de bits. Cela est confirmé par l'examen des 4 histogrammes associés (Fig. 5.4). En effet, on remarque que l'histogramme de  $a_1$  est semblable à celui de  $I_n$  tandis que ceux des 3 images de contours sont des histogrammes monomodaux avec une pointe accentuée en 0.

Par la suite, pour avoir une représentation hiérarchique sur plusieurs niveaux (disons  $J$  niveaux), il suffit de répéter la décomposition sur la sous-image  $a_1$ . Autrement dit,  $a_1$  sera "remplacée" par les 4 sous-images  $a_2, v_2, h_2$  et  $d_2$  puis  $a_2$  par  $a_3, v_3, h_3$  et  $d_3$  jusqu'à  $a_J, v_J, h_J$  et  $d_J$ . La représentation multirésolution est l'ensemble des sous-images  $a_H, \{v_j, h_j, d_j\}$  pour  $j = 1, \dots, J$ .

La reconstruction a lieu au niveau du décodeur. C'est l'image  $a_J$  qui est d'abord affichée. Un affichage de meilleure résolution passera par l'envoi de  $v_J, h_J$  et  $d_J$  puis la reconstruction de  $a_{J-1}$  via la transformation en ondelettes inverse. Là encore, on répète cette procédure jusqu'à l'obtention de  $I_n$  à partir de  $a_1, v_1, h_1$  et  $d_1$ . Au total,  $J$  passes suffiront pour reconstruire **exactement**  $I_n$ . Bien entendu, la reconstruction n'est pas parfaite si l'utilisateur effectue moins de  $J$  passes.

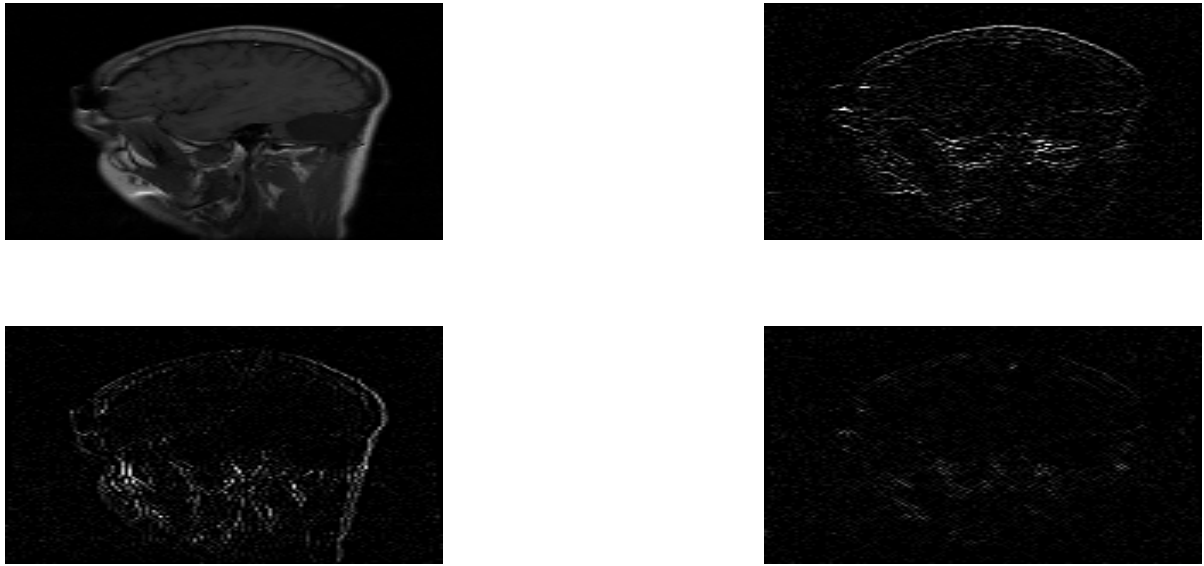


FIG. 5.3: Du haut vers le bas de la gauche vers la droite : images  $a_1$ ,  $h_1$ ,  $v_1$  et  $d_1$  obtenues après décomposition en ondelettes 5/3 de l'image "mri.8bit". Les sous-images des détails ont été amplifiées d'un facteur 10.

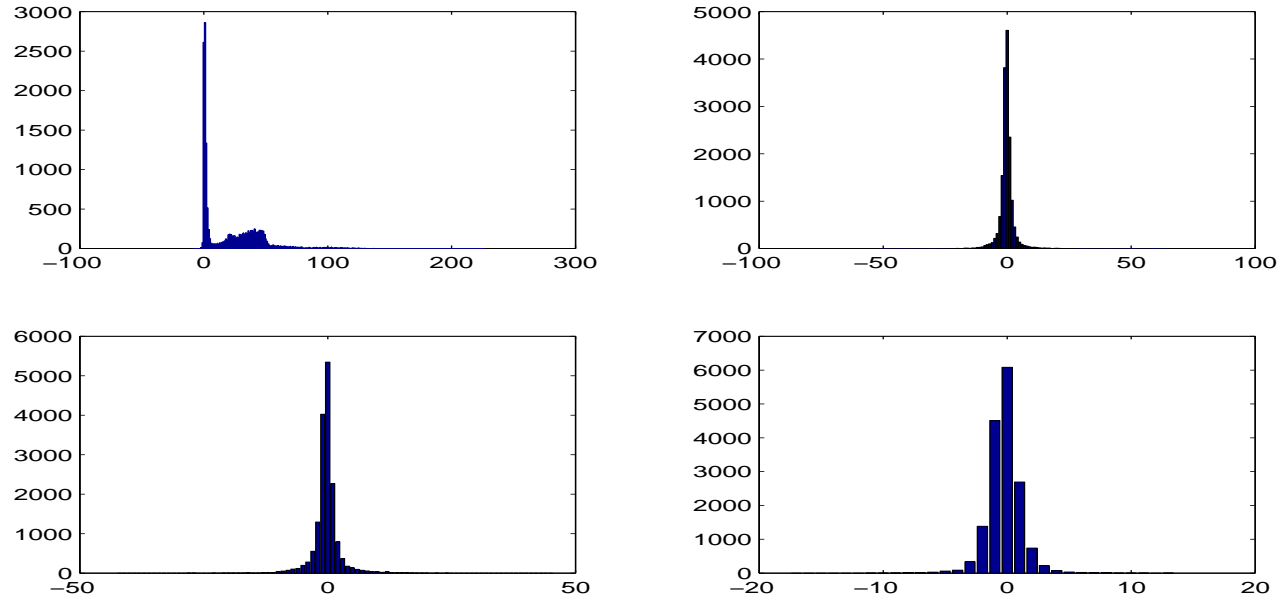


FIG. 5.4: Du haut vers le bas de la gauche vers la droite : histogrammes de images  $a_1$ ,  $h_1$ ,  $v_1$  et  $d_1$  obtenues après décomposition en ondelettes 5/3 de l'image "mri.8bit". Les sous-images des détails ont été amplifiées d'un facteur 10.

## 5.4.2 Codeur entropique

Il s'agit de coder en binaire les sous-images issues de la transformation en ondelettes. Pour ce faire, on utilise un codeur par **plans de bits** (*bit-plane encoding*). Supposons que la représentation binaire des intensités des pixels d'une sous-image se fasse sur  $B$  bits. Les plans de bits de cette sous-image sont les  $B$  images **binaires** de même taille telles que le plan d'indice  $b$  contient le bit d'indice  $b$  dans la représentation binaire de chaque intensité. Illustrons ceci par un exemple simple pour lequel  $B = 3$ . On considère que le début d'une ligne de l'image est : 5, 7, 0 et 3. Les représentations binaires sont donc : 101, 111, 000 et 011. On génère donc 3 images binaires. Le plan binaire d'indice 3 contient les bits les plus forts soit 1, 1, 0 et 0. Le plan binaire d'indice 2 contient les bits d'indice 2 c.à.d. 0, 1, 0 et 1. Enfin, le plan de bits d'indice 1 correspond aux bits les plus faibles à savoir 1, 1, 0 et 1. Sur la base de cet exemple didactique, il est possible de concevoir un algorithme simplifié permettant l'extraction des plans de bits de chaque sous-image de la hiérarchie. L'intérêt est que les plans d'indices élevés correspondent à une version **binaire** simplifiée de la sous-image considérée. Il est possible d'utiliser le savoir-faire acquis en matière de codage d'images binaires. A ce titre, c'est le codeur de la norme JBIG2 (le MQ codeur arithmétique) qui a été retenu pour la norme JPEG2000.

## 5.5 Premiers résultats

### 5.5.1 Versions préliminaires

A l'heure actuelle, il existe des exécutables de la partie I, accessibles gratuitement sur Internet. On citera notamment la "beta" version 1.0.0.1 du codeur baptisé **Image Power's JPEG2000 codec**<sup>1</sup>. Cette version émane de la proposition d'implémentation logicielle Jasper.

### 5.5.2 Complexité

La complexité opératoire de JPEG2000 est bien plus importante que celle des deux normes précédentes. Bien entendu, c'est le prix à payer pour aboutir à de meilleurs taux de compression.

### 5.5.3 Amélioration des taux de compression

Les images compressées par JPEG2000 requièrent moins de place mémoire que les images compressées par JPEG en mode sans pertes. On peut résumer les résultats préliminaires dans le tableau 5.1.

Débit (bit/pixel)	Facteur de réduction /JPEG sans pertes
0.25	53%
0.5	36%
0.75	18%
1	11%

TAB. 5.1: **Facteur moyen de réduction des tailles d'images compressées entre JPEG2000 et JPEG en mode sans pertes.**

---

<sup>1</sup>On y accède à l'adresse suivante <http://www.imagepower.com>. Tout renseignement sur cette version peut être obtenu dans le site suivant : <http://www.ece.ubc.ca/~mdadams/jasper>.

# Bibliographie

- [1] S. Wong, L. Zaremba, D. Gooden, H.K. Huang, "Radiologic Image Compression : A Review," *Proceedings of the IEEE*, Vol. 83, No. 2, pp. 194-219, February 1995.
- [2] A. Moffat, T.C. Bell, I.H. Witten, "Lossless Compression for Text and Images," *International Journal of High Speed Electronics and Systems*, Vol. 8, No. 1, pp. 179-231, March 1997.
- [3] CCITT Recommendation T4, *Facsimile Coding Schemes and Coding Control Function for Group 3 Facsimile Apparatus*, Malaga, Torremolinos, 1984.
- [4] CCITT Recommendation T6, *Facsimile Coding Schemes and Coding Control Function for Group 4 Facsimile Apparatus*, Malaga, Torremolinos, 1984.
- [5] CCIT, Draft Recommendation T.82 & ISO DIS 11544 : *Coded Representation of Picture and Audio Information - Progressive Bi-Level Image Compression*, 1993.
- [6] JPEG, ISO/IEC JTC1/ 10918-1 (1994) : *Information Technology- Digital Compression and Coding of Continuous-Tone Still Images : Requirements and Guidelines*.
- [7] G.K. Wallace, "The JPEG Still Picture Compression Standard", *Communications of the ACM*, Vol. 34, No. 4, pp. 31-44, April 1991.
- [8] M. Weinberger, G. Seroussi, G. Sapiro, " Low Complexity, Context-based, Lossless Image Compression Algorithm," *Proceedings of the Data Compression Conference*, Snowbird, Utah, USA, pp. 140-149, March 1996, <http://www.hpl.hp.com/loco>.
- [9] JPEG-LS, ISO/IEC JTC1/SC29 WG1 (1998), "Information Technology- Lossless and Near-lossless Compression of Continuous-Tone Still Images," Final Draft International Standard FDIS14495-1. Also Recommendation ITU T.87.
- [10] S.W. Golomb, "Run-length encodings," *IEEE Transactions on Information Theory*, Vol. IT-12, pp. 399-401, 1966.