

Une synthèse sur les formats usuels d'images numériques fixes

Michel Chilowicz

1 Introduction

Nous discutons ici brièvement des principaux formats utilisés couramment pour la représentation d'images numériques fixes en deux dimensions, exprimables par une matrice de pixels (images dites *bitmap*). Nous ne nous intéresserons cependant pas aux images vectorielles (ce qui ne nous empêchera pas d'évoquer des formats de description de pages pouvant intégrer des images bitmap).

Dans le cadre de cette synthèse, nous classons les formats d'image en quatre catégories :

- Les formats bruts : ils définissent une simple convention pour l'expression de la matrice ; aucune technique de compression n'est envisagée.
- Les formats compressifs sans perte : ces formats réalisent une compression sur la matrice image. La transformation entre un format brut et un format compressif sans perte est bijective.
- Les formats compressifs destructifs : ils permettent d'atteindre de meilleurs taux de compression au prix d'une dégradation de l'image. Ils utilisent généralement une quantification sur une transformée fréquentielle, telle que la transformée en cosinus discret pour JPEG et la transformée en ondelettes pour JPEG2000.
- Les formats composites : ces formats peuvent être considérés comme des conteneurs pouvant accueillir notamment de multiples images fixes. Les formats de description de pages entrent dans cette catégorie.

En pratique, les formats d'image peuvent généralement être distingués par leur en-tête caractéristique (avec présence de *mot magique* par exemple), par l'extension des noms de fichiers, voire par l'usage d'un type MIME spécifique¹. Ces moyens de distinction permettent la sélection du visualisateur adéquat pour un fichier (par exemple pour un navigateur Web qui utilisera le type MIME renvoyé par le serveur HTTP).

Plutôt que d'entreprendre un inventaire exhaustif des formats d'image numérique (ce qui serait tâche impossible en si peu de pages), nous abordons les différentes caractéristiques majeures nécessaires pour qualifier un format : nous en profitons pour citer quelques formats d'images usuels au passage. Le lecteur pourra trouver en fin de document un index répertoriant tous les formats cités ainsi que des références vers les spécifications de ces formats.

¹Les images bitmap sont de type MIME *image* et de sous-type définissant leur format : *jpeg* pour le format JPEG [13] ou *png* pour le PNG [23]. Un registre des types et sous-types MIME [15] est maintenu par l'IANA [9].

2 Espaces colorimétriques

Un pixel (unité élémentaire de la matrice image) est caractérisée par sa couleur. La couleur est une perception lumineuse réceptionnée par l'oeil² et interprétée par le cerveau : cette perception s'avère différente d'un animal à l'autre voire au sein d'un groupe d'individus d'une même espèce³. Une couleur correspond physiquement à un mélange de lumières réémises de longueurs d'ondes déterminées : on propose donc des bases colorimétriques pour représenter les couleurs visibles par l'humain.

2.1 Images binaires

Une image numérique binaire se caractérise par des pixels codés sur un bit définis par deux valeurs : 0 ou 1. L'utilisation d'images binaires peut par exemple être utilisée pour la représentation de documents scannés en noir et blanc ou alors pour des télécopies. Les formats couramment utilisés sont le PBM [24] (de la famille de formats PNM [24]) ou le TIFF [28] g3 pour les télécopies (avec un codage compressif RLE). On peut également citer le format WBMP [29] (désormais obsolète) conçu pour le protocole WAP.

2.2 Images monochromes

Il est possible de coder un pixel par une seule composante de luminosité, par exemple pour représenter des images en niveaux de gris. Les bâtonnets, cellules spécialisées dans la perception de la luminosité de l'oeil humain, ne permettent de distinguer qu'environ 200 intensités différentes. C'est pourquoi on utilise généralement une profondeur de 8 bits (256 valeurs) pour coder un pixel monochromatique.

2.3 Synthèse additive en trichromie

2.3.1 La base RVB (Rouge Vert Bleu)

Pour la plupart des applications numériques consistant à représenter des images sur un écran, la base trichromique de synthèse additive RVB (Rouge Vert Bleu) est utilisée (les écrans à tube cathodiques ou à diodes électro-luminescentes utilisent ces trois couleurs primaires pour la synthèse de couleurs visibles par l'homme). Chaque couleur y est alors représentée par un triplet de réels dans $[0, 1]$ (le triplet $(1, 1, 1)$ étant la couleur blanche et $(0, 0, 0)$ la couleur noire). La quasi-totalité des formats d'images permettent de gérer des images en base RVB où chaque composante est généralement codée sur 8 bits (24 bits pour un pixel). Cependant cette base présente l'inconvénient de ne pas être standardisée : on définit à cet effet un profil ICC (*International Colour Consortium*) [10] réalisant une bijection⁴ entre la base RVB du périphérique d'affichage et une base colorimétrique standardisée telle que CIE-L*a*b* ou CIE-XYZ. Il existe toutefois des tentatives de normalisation de bases RVB telles que sRGB [44] ou Adobe RGB [2].

Quelques formats bruts en base RVB Nous citons ici quelques exemples de formats bruts utilisant le RVB (cette liste n'a nullement vocation à être exhaustive) et stockant les pixels séquentiellement, ligne par ligne (un en-tête étant présent pour spécifier la taille et la profondeur de l'image) :

- PPM [24]. Ce format utilise une profondeur par composante de 8 ou 16 bits ; les composantes peuvent également être codées en ASCII. Des variantes pour les images en niveau de gris (PGM) et binaires (PBM) existent.

²Chez l'espèce humaine, la perception colorimétrique est assurée par les cônes, plus sensibles en haute luminosité.

³Par exemple, au sein de l'espèce humaine, certains individus peuvent présenter des anomalies de vision des couleurs appelées dyschromatopsies (ou daltonisme).

⁴Cette bijection est obtenue par un procédé d'étalonnage.

- BMP [3]. Ce format créé par Microsoft permet également le traitement d'images en niveaux de gris et propose une compression RLE.
- TGA [27]. Il propose une compression RLE
- Rasterindex . Ce format proposé par SUN permet un codage brut ou une compression RLE.
- XPM. Ce format présente la particularité de se présenter sous la forme de fichier source C (image conservée dans un char *). Des images XPM peuvent donc être facilement compilées au sein d'un programme C ou C++. Ce format est principalement utilisée avec le système X-Window pour définir des curseurs et icônes : son format ASCII le rend peu adapté pour des images volumineuses. Une variante XBM existe pour les images binaires.

2.3.2 Les bases teinte-saturation-valeur de la CIE

Ces bases ne sont généralement pas utilisées dans les formats d'image, elles s'avèrent néanmoins plus intuitives à manipuler en prenant en compte les propriétés du système visuel humain . On notera que la Commission Internationale de l'Éclairage (CIE) [4] propose des bases teinte-saturation-valeur standardisées :

- La base CIE-L*a*b* (luminance a b).
- La base CIE-L*u*v*.
- La base CIE-XYZ.

Il existe des formules spécifiques pour assurer la conversion d'une base à une autre (que nous ne citerons pas ici).

2.3.3 Les bases luminance-chrominances

Les bases luminance-chrominances sont utilisés pour la transmission analogique de vidéos (par exemple YUV est la base utilisée pour la norme européenne PAL et YIQ pour la norme américaine NTSC). Les normes JPEG [13] et MPEG [17] utilisent une variante de YUV, YC_bC_r , avec sous-échantillonnage des composantes de chrominance.

2.4 Synthèse soustractive : base CMJ et CMJN

Pour l'imprimerie, il est nécessaire d'utiliser une base colorimétrique de synthèse soustractive. On utilise couramment la base quadrichromique Cyan Magenta Jaune Noir (CMJN) : en théorie les composantes cyan, magenta et jaune forment déjà une base (le noir pouvant être obtenu par un triplet (1, 1, 1)). En pratique, l'utilisation fréquente de la couleur noire pour l'impression de textes, et l'obtention d'un noir de mauvaise qualité par mélange d'encre ont conduit à l'ajout d'une composante noire. Tout comme la base RVB, la base CMJN n'est pas standardisée et est spécifiques aux dispositifs d'impression. Ainsi la conversion d'une image en base RVB vers une base CMJN peut être réalisée en connaissant les profil ICC de la base RVB et de la base CMJN en passant par la base absolue pivot CIE La^*b^* ou XYZ.

Certains formats tels que le TIFF [28] supportent la base quadrichromique CMJN ou le format standardisé PDF/X de description de page pour la pré-presse.

2.5 Canal de transparence

Certains formats d'image proposent, en plus des traditionnels canaux utilisés pour le codage des composantes chromatiques, un canal spécifique de transparence (appelé canal *alpha*). Celui-ci est utilisé pour la superposition et le mélange d'images. Il peut également être utilisé pour définir des images non-rectangulaires en spécifiant des zones transparentes.



Figure 1: Images à couleurs indexées de Lenna (la conversion de l'espace RVB vers un espace de couleurs indexées est réalisée en utilisant l'algorithme de quantification décrit sur <http://www.imagemagick.org/script/quantize.php>).

Le codage du canal de transparence peut être binaire (pixel transparent⁵ ou non) comme pour le format GIF [8] utilisé sur le Web ou alors avoir une profondeur supérieur à 1 bit comme pour le format PNG [23] où la transparence peut être codée comme une composante chromatique (sur 8 ou 16 bits).

2.6 Formats bruts pour capteurs CCD

La plupart des formats bruts utilisés par les fabricants de capteurs CCD (utilisés par les scanners et appareils photographiques numériques) sont propriétaires et peu documentés. Ces formats expriment les données brutes obtenues par le capteur CCD avant toute opération de post-traitement nécessaire à leur conversion en format usuel utilisant une base RVB. L'utilisation de formats bruts est intéressante afin d'affiner certains réglages pour le post-traitement (saturation de la couleur, balance des blancs, ...) ce qui est quelquefois préférable à la conversion automatique en format RVB (généralement en JPEG) opérée par le matériel.

Format DNG On peut citer l'initiative d'Adobe proposant en 2004 une première version de la spécification DNG [5] (*Digital Negative Specification*) visant à créer un standard pour les fichiers de données brutes de capteurs CCD. Ce format est à l'heure actuelle supporté par un nombre très limité de matériels.

2.7 Couleurs indexées

Nous nous sommes intéressés jusqu'à présent aux images naturelles définies par des couleurs dont les valeurs chromatiques sont des valeurs réelles (discrétisées pour les besoins de la numérisation). Pour certaines images de synthèse, il peut être utile d'utiliser un nombre limité de couleurs spécifié par une palette. Les formats GIF [8], PNG [23] ou TIFF [28] offrent la possibilité de réaliser un codage des couleurs par palette, généralement limitée à 256 valeurs (8 bits). Ce procédé utile pour les images de synthèse comportant un nombre limité de couleurs peut également être utilisé sur des photographies (voir figure 1) comme procédé de compression destructif par quantification (un dictionnaire de couleurs est composé afin de limiter la destruction et chaque couleur réelle est remplacée par la valeur la plus proche du dictionnaire).

⁵Si le pixel est transparent, il est normalement inutile de spécifier des composantes chromatiques.

3 Transmission et stockage des matrices de pixels

L'organisation du stockage de la matrice de pixels pour un format peut être étudiée afin de permettre un affichage dégradé de l'image par la transmission des premiers octets du fichier. Nous analysons ici différentes méthodes d'organisation de matrices de pixels.

3.1 Organisation séquentielle

La plupart des formats de fichiers organisent le stockage de la matrice de pixels sous forme séquentielle : les pixels sont sérialisés ligne par ligne (ou colonne par colonne), voire pour certains formats bloc par bloc (c'est le cas du format JPEG [13] où les images sont subdivisées en blocs de 8x8 pixels). Il est possible de procéder à une séparation des canaux, notamment pour leur appliquer une compression indépendante.

3.2 Aperçus

Pour certaines images volumineuses, il peut être avantageux de disposer d'un aperçu en dimensions réduites⁶ pour visualisation de planches d'images d'un catalogue. Une approche classique consiste à créer des aperçus en fichiers séparés. Une approche préférable serait l'inclusion en début du fichier image en dimensions originales d'un aperçu de dimensions réduites. L'intégration d'aperçus est prévue notamment pour les formats d'image TIFF [28] et JPEG [13]. On peut citer également le format Flashpix [7] proposant de multiples niveaux d'aperçus.

3.3 Organisation par plans de bits

Le découpage d'une image en plan de bits (découpage d'une composante colorimétrique de n bits en n plans de bits indépendants) permet d'obtenir des taux de compression non-destructive plus intéressants sur les plans correspondant aux bits de poids fort. Il est possible alors d'indiquer les plans de bits dans le fichier, du plan de bit de poids fort au plan de bit de poids faible : une application de visualisation réceptionnant les premiers plans de bits peut reconstituer une approximation de l'image. Dans certains cas, l'absence des plans de bits de poids faible n'est pas perceptible par l'oeil humain : il s'agit alors d'une forme de compression destructrice par quantification.

3.4 Entrelacement

La transmission par entrelacement d'une matrice de pixels consiste à fractionner en plusieurs passes la spécification de la matrice : pour chaque passe, certains pixels de la matrice sont transmis. Il est possible alors de reconstituer une image approchée (par interpolation) par la connaissance d'une ou plusieurs premières passes. La transmission de toutes les passes permet de reconstituer l'image sans dégradation.

Le procédé d'entrelacement le plus courant est *Adam7* où la transmission d'une matrice se réalise en 7 passes par découpage de l'image en blocs de 8x8 pixels ; chaque passe consistant à la transmission de certains pixels (voir la figure 2). Ce procédé est utilisé par les formats TIFF [28], PNG [23] et GIF [8].

⁶Divers algorithmes de réduction de taille d'image peuvent être utilisés comme la sélection de pixels de l'image, le moyennage de pixels ou alors l'utilisation d'une transformée fréquentielle voire en ondelettes. On notera que la réduction de taille d'image peut poser problème pour des images en palettes : le passage dans une base non-indexée peut être nécessaire.

$$\begin{bmatrix} 1 & 6 & 4 & 6 & 2 & 6 & 4 & 6 \\ 7 & 7 & 7 & 7 & 7 & 7 & 7 & 7 \\ 5 & 6 & 5 & 6 & 5 & 6 & 5 & 6 \\ 7 & 7 & 7 & 7 & 7 & 7 & 7 & 7 \\ 3 & 6 & 4 & 6 & 3 & 6 & 4 & 6 \\ 7 & 7 & 7 & 7 & 7 & 7 & 7 & 7 \\ 5 & 6 & 5 & 6 & 5 & 6 & 5 & 6 \\ 7 & 7 & 7 & 7 & 7 & 7 & 7 & 7 \end{bmatrix}$$

Figure 2: Matrice d'entrelacement *Adam7*. Lors de la première passe, le pixel en haut à gauche de chaque bloc 8x8 est transmis. Lors de la dernière passe, les lignes impaires sont transmises.

3.5 Organisation pyramidales par blocs

Une image peut être transmises progressivement par transmission de vignettes de dimensions croissantes. Pour la passe n_i , on découpe l'image en blocs de taille $n_i \times n_i$, pour la passe n_{i+1} , en blocs de taille $\frac{n_i}{2} \times \frac{n_i}{2}, \dots$: pour chacun de ces blocs, un pixel représentatif est conservé⁷. Plutôt que de transmettre les valeurs absolues de chaque pixel, il est possible de réaliser un codage différentiel avec les pixels de la passe précédente. Ce procédé (adapté à une décomposition fréquentielle de l'image) est utilisé par JPEG [13] et JPEG2000 [14].

4 Compression sans perte

Une image sous la forme de matrice de pixels comporte de nombreuses redondances dans sa structure (plage de pixels uniforme, motifs répétés, ...) qui peut bénéficier d'algorithmes de compression généralistes. La limite théorique des performances de compression sans perte est définie par l'entropie de la source : l'obtention de meilleurs taux de compression nécessite l'utilisation de techniques de compression avec perte dégradant l'information par des procédés de quantification ou de transformation.

Nous décrivons ici très brièvement quelques algorithmes de compression usuels.

4.1 Opérations préalables à la compression : découpage en plans et codage différentiel

4.1.1 Découpage en plans

Préalablement à l'utilisation d'une méthode de compression sans perte, il peut être judicieux de réaliser une opération de décorrélation de l'image. Il est possible, par exemple, de procéder à la compression séparée des différents plans colorimétriques de l'image (canaux **R**, **V** et **B** par exemple), voire au découpage en plans binaires (par exemple, une image en niveaux de gris de 8 bits peut être découpée en 8 plans binaires pour chacun des bits).

4.1.2 Codage différentiel

Un procédé de codage différentiel peut également être utilisé : un tel procédé ne code plus les pixels individuellement mais comme différence par rapport aux pixels du voisinage. On peut par exemple utiliser une technique de prédiction par interpolation linéaire d'un pixel à partir de son voisinage et ne coder que la différence avec la prédiction réalisée (codage ADPCM). Un codage

⁷Par exemple le pixel en haut à gauche, le pixel moyenne ou médian.

différentiel prédictif est notamment utilisé par le format de fichier PNG [23] préalablement à une compression LZ77.

4.2 Codage de Huffman : codage optimal à longueur variable

Le codage de Huffman [42] permet de réaliser un codage préfixe optimal à longueur variable à partir d'un alphabet dont la fréquence d'apparition des lettres est connu. Une première passe consiste donc à relever la fréquence des lettres ; lors de la deuxième passe un arbre est construit en commençant par ses feuilles. À partir de l'arbre de Huffman obtenu, on associe à chaque branche un bit afin de créer un code préfixe. Il existe également des codages de Huffman dynamiques ne nécessitant pas la première étape préalable de comptabilisation de fréquences.

4.3 Codage RLE

Certains fichiers images comportent des suites de pixels similaires (ou des valeurs similaires sur un plan), particulièrement des images de synthèse à couleurs indexées : le principe du codage RLE (*Run Length Encoding*) consiste alors à coder une suite de valeurs par couples (valeur, nombre d'occurrences consécutives de la valeur).

Le codage RLE est notamment utilisable par les formats de fichiers BMP [3], PCX [18] ou TIFF [28].

4.4 LZ77 : compression par répétitions sur fenêtre glissante

Principe L'algorithme de compression LZ77 [47] fonctionne par recherche de répétitions sur une fenêtre glissante de taille fixe : cette fenêtre, centrée sur la position courante se découpe en deux parties, à gauche l'historique et à droite un tampon du texte à examiner. Le plus grand préfixe à partir de la position courante pouvant être trouvée dans la portion historique est pris en compte, puis codé par sa position dans la fenêtre et sa longueur. Si aucun préfixe ne peut être trouvé, le caractère courant est codé littéralement (ou en utilisant un codage de Huffman). LZ77 s'avère efficace pour la compression de motifs répétés localement.

Utilisation L'algorithme LZ77 est utilisé par l'outil de compression *gzip* [33], la librairie correspondante *zlib* [40] ainsi que par le format d'archives *ZIP*. Il est employé par le format PNG [23] après codage différentiel prédictif.

4.5 LZ78 : compression par dictionnaire

Principe L'algorithme LZ78 [48] construit dynamiquement lors de l'examen des données un dictionnaire de motifs. Chaque entrée du dictionnaire est un couple (i, j) où i est une référence vers une précédente entrée du dictionnaire et j le caractère ajouté au mot désigné par l'entrée i . Il est nécessaire de fixer une taille limitée pour le dictionnaire, par exemple en le vidant périodiquement ou alors en éliminant certaines entrées correspondant à des motifs peu fréquents. LZ78 présente néanmoins l'avantage, par rapport à LZ77, de pouvoir considérer des motifs répétés non-nécessairement locaux.

Variante LZW Une variante de LZ78, LZW [46], permet d'éviter le codage du caractère j (caractère suivant un motif du dictionnaire). LZW est utilisé par l'outil de compression *compress* ainsi que certains formats d'images tels que GIF [8] ou TIFF [28]. Cet algorithme de compression faisant l'objet d'un brevet logiciel d'Unisys (expiré en 2004) a rendu son utilisation moins populaire que LZ77.

4.6 Compression par transformée de Burrows-Wheeler

Principe de la transformée de Burrows-Wheeler Burrows et Wheeler ont proposé en 1994 une transformation pour des séquences de caractères sur un alphabet. Soit v un mot sur A^* (A étant un alphabet ordonné), sa transformée de Burrows-Wheeler est obtenue par les dernières lettres de ses conjugués triés par ordre lexicographique. Il est possible de montrer que cette transformée est injective : à partir de la séquence des dernières lettres des conjugués triés et de l'indice correspondant au conjugué v , il est possible de reconstituer v (la transformée pouvant être interprétée combinatoirement comme une permutation sur $1, \dots, |v|$). La réversibilité de cette transformée permet de l'utiliser pour la compression sans perte.

Application à la compression Les conjugués d'un mot possédant un long préfixe commun correspondent à un facteur du mot répété à plusieurs reprises : il est probablement extensible d'un caractère à gauche (ce qui correspond à la dernière lettre des conjugués). Par exemple les conjugués de préfixe *he·* ont une forte probabilité de se terminer par *t* (facteur *the·*). On en déduit que la transformée de Burrows-Wheeler permet de regrouper des caractères identiques d'où son intérêt pour la compression.

Logiciel de compression bzip2 Le logiciel de compression bzip2 [31] utilise la transformée de Burrows-Wheeler sur des blocs et applique un codage *Move To Front* puis un codage de Huffman. Les performances obtenues sont excellentes pour les textes en langue naturelle mais également pour des images.

4.7 Un comparatif rapide

Nous réalisons un comparatif rapide des algorithmes de compression généralistes décrits précédemment sur un jeu de trois images utilisant un format brut RVB ou niveaux de gris (avec une profondeur par composante de 8 bits). Nous utilisons pour établir ce comparatif les différents formats de fichiers ainsi que les algorithmes de compression généralistes sur les formats PPM et PGM (formats bruts avec en-tête) suivants :

- Format de fichier BMP avec compression RLE.
- Format de fichier PNG avec codage différentiel et compression LZ77.
- Utilisation de *gzip* [33] (LZ77) sur formats PPM et PGM.
- Utilisation de *compress* (LZW) sur formats PPM et PGM.
- Utilisation de *bzip2* [31] (transformée de Burrows-Wheeler, MTF et Huffman) sur formats PPM et PGM.

Les résultats sont spécifiés en figure 3.

On constate que la compression RLE mise en oeuvre par BMP n'est efficace que pour les images comportant d'importantes plages uniformes (comme *gray*) : pour des images photographiques RLE entraîne un surcout (il n'y a quasiment pas de pixels identiques consécutifs). LZ77 permet en général de meilleures performances que LZW. *bzip2* permet d'obtenir les meilleurs taux de compression : il n'est cependant que peu utilisé par des formats d'images sans perte. Cela peut s'expliquer par la conception récente de l'algorithme (1994) et les ressources importantes nécessaires à la compression et à la décompression (notamment mémorielles).

5 Compression avec perte

L'utilisation de méthodes de compression avec perte s'avère très utile pour réduire la taille d'images telles que les images issues d'un signal réel (photographies). On cherche alors à diminuer l'entropie de l'image tout en minimisant la dégradation perceptible par l'homme (mesure très subjective). On peut utiliser comme mesure de distorsion objective l'erreur quadratique moyenne sur les pixels.




			
	Lenna	Boat	Gray21
	512x512, RVB	512x512, gris	512x512, gris
PNM (sans compression)	786447	262159	262159
BMP (RLE)	-	496650	5176
PNG (LZ77)	734430	218602	1436
gzip -best (LZ77)	733270	217948	1256
compress 16 bits (LZW)	832533	241185	6221
bzip2 -best (BW)	584483	188777	221

Figure 3: Tailles des fichiers pour différents procédés de compression d'image sans perte.

5.1 Techniques de quantification

Principe Un signal lumineux peut être physiquement exprimée dans un R-espace vectoriel à trois dimensions (en utilisant une base RVB, $L^*a^*b^*$, ...) : sa numérisation nécessite une étape de quantification ; cette technique utilise une fonction surjective afin d'associer chaque valeur à une classe d'équivalence la représentant. Celle-ci permet d'obtenir un triplet de valeurs entières, chacune codée sur k bits (k étant la profondeur). Il peut également être nécessaire de diminuer la profondeur d'une composante en appliquant une nouvelle quantification. Lors de l'utilisation d'une quantification, on cherche à minimiser la distorsion appliquée au signal (mais également la maximisation du taux de compression pouvant être obtenu après quantification).

Méthodes de quantification En l'absence de données sur l'image, une quantification classique consiste à utiliser une fonction linéaire : l'espace de départ est découpé en intervalles de taille fixe entre deux valeurs extrêmes, chaque valeur avant quantification étant représentée par un de ces intervalles. Une autre possibilité, en connaissant l'histogramme de l'image, consiste à utiliser des intervalles de taille variable : les intervalles sont de taille importante lorsqu'ils contiennent peu de valeurs de l'image et de taille faible dans les régions à forte densité de valeurs. Il existe des méthodes algorithmiques telle que l'algorithme de Lloyd-Max permettant de déterminer ces intervalles en minimisant la distorsion. Il existe également des procédés de quantification vectorielle où l'image est divisée en blocs de n blocs et est considérée à valeurs dans un espace vectoriel de dimension n^2 .

5.2 Transformée en cosinus discret (DCT)

5.2.1 La transformée

La transformée en cosinus discret (DCT) est une transformée fréquentielle (comparable à la transformée de Fourier discrète en certains points) utilisée par le format JPEG [13]. L'image est découpée en blocs de taille $N \times N$ qui sont symétrisés horizontalement et verticalement pour l'élimination de hautes fréquences parasites liées aux frontières, l'obtention des coefficients de la matrice DCT utilise la formule suivante (où $C(u)$ et $C(v)$ sont des coefficients normalisateurs

définis par $C(0) = \frac{1}{\sqrt{2}}$ et $C(u \neq 0) = 1$:

$$y_{u,v} = DCT(x_{k,l}) = \frac{2}{N} C(u) C(v) \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} \cos\left(\frac{(2k-1)\pi u}{2N}\right) \cos\left(\frac{(2l-1)\pi v}{2N}\right) x_{k,l} \quad (1)$$

La matrice obtenue est à valeurs réelles : les valeurs sont élevées sur la première ligne et la première colonne (fréquences basses) et décroissent vers le coin inférieur droit (fréquences élevées). La répartition du signal engendrée par la DCT est adaptée à la vision humaine beaucoup plus sensible sur les basses fréquences.

5.2.2 Algorithme

Naïvement, la DCT est réalisée en temps $O(N^2)$: il existe cependant des algorithmes (que nous ne détaillerons pas) bien plus efficaces afin de réaliser une transformée en temps $O(n \log_2(n))$ [41].

5.2.3 Quantification

Il est ensuite nécessaire d'entreprendre une quantification des valeurs réelles de la matrice obtenue par DCT. On pourra à cet effet ne pas conserver certains coefficients vers le coin inférieur droit de la matrice et se contenter de coder les coefficients supérieurs gauches (sélection zonale), ou ne conserver que les coefficients supérieurs à un seuil (sélection par seuil). Une autre solution, préconisée par le format JPEG⁸, consiste à utiliser une matrice de quantification spécifiant les seuils pour chaque coefficient puis une quantification scalaire uniforme. Néanmoins cette solution nécessite l'inclusion de la matrice dans le fichier produit afin de pouvoir réaliser le décodage.

5.2.4 Transformée inverse

Afin de retrouver la matrice de pixels (dégradée) de l'image, il est nécessaire de réaliser une DCT inverse.

5.2.5 Processus de codage JPEG

Voici les principales étapes du codage d'une image en utilisant le standard JPEG :

1. L'image en base RVB est convertie en base $Y C_b C_r$.
2. Les composantes de chrominance (C_b et C_r) sont sous-échantillonnées⁹. Ensuite les trois composantes sont traités séparément.
3. Pour chaque composante, on groupe les valeurs par blocs de 8x8 valeurs sur lesquels on réalise une DCT.
4. La matrice obtenue par DCT est quantifiée (ce qui induit une compression destructrice de l'image).
5. Pour chaque composante, on entreprend une compression sans perte des matrices quantifiées en utilisant un codage entropique : celui-ci consiste à parcourir les blocs de l'image en zigzag puis à utiliser une compression RLE et un codage de Huffman (voire un codage arithmétique).

Le stockage de données compressées par la méthode JPEG dans un fichier est définie par la norme JFIF [12] (*JPEG File Interchange Format*).

⁸Le format JPEG n'impose pas de méthode de quantification : l'implantation du codeur est libre. Cependant une image codée doit pouvoir être décodée avec n'importe quel décodeur conforme.

⁹En pratique, on utilise soit une sous-échantillonnage de facteur 2 uniquement horizontalement, soit horizontalement et verticalement.

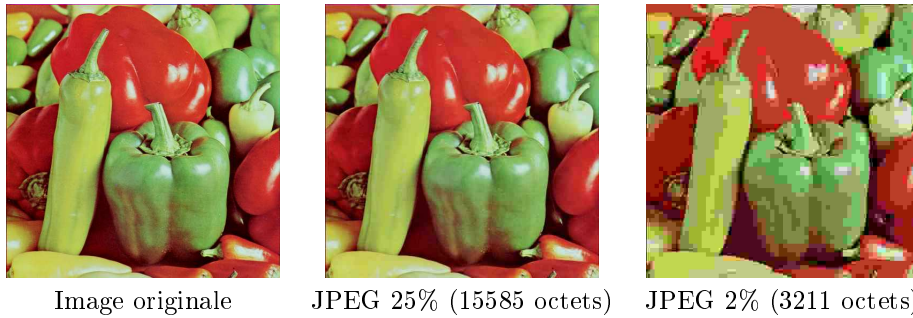


Figure 4: Compression JPEG de *peppers* (512x512, RVB, 8 bits par composante, soit 786432 octets)

5.3 Compression par ondelettes

5.3.1 Transformée en ondelettes

La transformée en ondelettes est une transformée mathématique de fonction, qui contrairement à la transformée de Fourier, permet d'obtenir des composantes localisées à la fois dans le domaine temporel et fréquentiel.

La transformée en ondelettes s'obtient ainsi (s et τ étant des coefficients de translation et dilation que l'on fait varier afin de couvrir le plan temps-fréquence) :

$$\gamma(s, \tau) = \int_{-\infty}^{+\infty} f(t) \psi_{s,\tau}^*(t) dt \quad (2)$$

Les ondelettes sont de la forme $\psi(s, \tau) = \frac{1}{\sqrt{s}} \psi\left(\frac{t-\tau}{s}\right)$: il est nécessaire de choisir ψ , l'ondelette mère (parmi les ondelettes mères classiques, il est possible de citer le *chapeau mexicain*, l'ondelette de Morlet ou l'ondelette de Haar).

Le procédé de compression par ondelettes réalise une analyse par ondelettes en multirésolutions : on découpe l'image originale en deux composantes avec des filtres passe-haut et passe-bas : l'une des composantes décrit l'allure générale de l'image, l'autre ses détails ; ces détails font l'objet d'une transformation par ondelettes et d'une quantification. Le procédé est ensuite répété pour une résolution demie de la précédente.

5.3.2 Le format JPEG2000

Processus de compression JPEG2000 Nous décrivons ici (très sommairement) les étapes du processus complexe de compression utilisé par JPEG2000 [14] :

- L'image est convertie de la base RVB en base $Y C_b C_r$.
- Les composantes de chrominance sont sous-échantillonnées.
- Pour chaque composante, l'image est découpée en régions (tuiles) : ce découpage est réalisé afin de réaliser des transformations en ondelettes indépendantes pour chacune de ces régions (pour des raisons d'économie de mémoire). Le découpage en tuiles est cependant à éviter car pouvant occasionner des frontières visibles sur l'images compressée.
- Une transformée en ondelettes est réalisée sur chaque tuile jusqu'à une profondeur arbitraire.
- Une quantification est réalisée sur les coefficients obtenus par la transformée en ondelettes.

- Les sous-bandes quantifiées sont découpées en zones (*precincts*).
- Les zones sont découpées elles-mêmes en blocs.
- Chaque plan de bits d'un bloc est codé en trois passes en utilisant l'algorithme EBCOT [45].
- Les bits sélectionnés par l'étape précédente font l'objet d'un codage arithmétique.
- Le résultat du codage est organisés en paquets, eux-mêmes empaquetés dans des couches¹⁰ : les couches sont placés dans le flux dans un ordre adéquat afin d'assurer une augmentation progressive de la qualité de l'image décodée lors du traitement de nouveaux paquets.

Fonctionnalités avancées de JPEG2000 Le format JPEG2000 propose outre un mode de compression par ondelettes, une possibilité de compression par DCT (comme le JPEG 1992). Nous citons ici quelques fonctionnalités avancées introduites par le format JPEG2000 :

- Utilisation de codes détecteur et correcteurs d'erreurs (JPWL).
- Utilisation d'algorithmes de chiffrement (JPSEC).
- Introduction de régions d'intérêt (ROI) bénéficiant d'une compression moins agressive.
- Possibilité de codage sans perte.
- Possibilité d'accès aléatoire à une zone de l'image (utile pour les images de dimensions conséquentes).
- Support de 256 canaux chromatiques (avec embarquement possible de profils ICC).
- Support d'un système de métadonnées.
- Support d'un canal de transparence.

5.4 Compression fractale

5.4.1 Principe

La compression fractale permet d'interpréter une image comme un ensemble de régions redondantes, une région pouvant être l'image d'une autre par une suite de transformations élémentaires (rotations, symétries, changement d'intensité, ...).

5.4.2 Méthode Jacquin

Une méthode simple de compression fractale a été proposée par Jacquin en 1992 [43]. Son principe consiste à découper avec deux pavages l'image à compresser : un pavage source et un pavage destination¹¹. Le pavage destination est plus dense (plus de blocs) que le pavage source. On recherche ensuite des couples (bloc source, bloc destination) similaires¹² après application de transformations telle que homothétie, symétrie, rotation et décalage d'intensité. La liste des couples et transformations associées est sauvegardée. Pour la décompression, on réalise les deux pavages et on applique plusieurs fois les transformations mentionnées par les couples pour converger après plusieurs itérations vers l'image originale.

¹⁰L'organisation des paquets dans les couches est laissée à la libre appréciation du codeur (aucun comportement n'est imposé).

¹¹Pour plus de simplicité, on peut choisir un pavage carré, d'autres formes étant envisageables.

¹²On cherche à minimiser l'écart-type des pixels des deux régions.

5.4.3 Format FIF

La société Iterated Systems a élaboré le format FIF de compression fractale : il s'agit de l'un des rares format de compression fractale existant. Il permet d'obtenir des taux de compression de 50:1 sans dégradation notable de la qualité d'image. D'autre part, un fichier compressé par méthode fractale ne contient qu'une liste de transformations mathématiques : il est possible de réaliser une décompression avec profondeur et résolution arbitraire (ce qui permet des agrandissements sans pixellisation). On notera que le format FIF fait l'objet d'un brevet ce qui limite considérablement son utilisation. D'autre part, la compression fractale nécessite des ressources calculatoires très largement supérieure à la compression par ondelettes pour des taux de compression presque similaires, ce qui est un sérieux handicap à sa diffusion.

6 Métadonnées

Historiquement, les formats d'images se sont contentés de stocker une image (matrice de pixels). Afin de faciliter la constitution de bibliothèques d'images, l'utilisation de métadonnées sur les images, telles que des informations sur les moyens techniques de numérisation, les droits d'auteur et des informations sur le contenu des images, s'avère indispensable. La plupart des formats de conception récente intègrent la gestion de métadonnées sur l'image. Le support le plus élémentaire de métadonnées consiste à autoriser un champ de commentaire pour l'image. Certains formats introduisent des champs prédéfinis ainsi que des champs définissables par l'utilisateur. On notera également que certains systèmes de fichiers permettent d'assigner des métadonnées à des fichiers : cependant celles-ci sont externes au fichier et sont généralement perdues lors d'une transmission. On ne s'intéresse donc qu'aux systèmes de métadonnées internes aux formats pour ensuite évoquer le cas des métadonnées internes à l'image (filigrane et stéganographie).

6.1 Quelques schémas de métadonnées usuels

6.1.1 IPTC

L'IPTC (*International Press Telecommunications Council*), un consortium international d'agences de presse, a élaboré un ensemble de champs de métadonnées [11] spécifiant notamment la provenance d'une image, des informations sur sa licence, sa description et des mots-clés s'y rapportant.

6.1.2 Exif

La spécification Exif (*EXchangeable Image File format*) a été proposée par la JEIDA (*Japan Electronic Industry Development Association*) et est désormais utilisée par la plupart des fabricants d'appareils photographiques numériques afin d'incorporer des métadonnées d'informations techniques sur la prise de vue (ouverture, obturation, balance des blancs, sensibilité, localisation pour les appareils dotés de GPS, date de prise de vue, ...). Les formats JPEG [13] et TIFF 6.0 [28] intègrent les métadonnées Exif.

6.2 Procédé d'encapsulation des métadonnées XMP

XMP (*eXtensible Metadata Platform*) est une spécification proposée par Adobe basée sur RDF (*Resource Description Framework*) (lui-même basé sur XML) pour l'intégration de métadonnées dans les fichiers images. Il existe certains schémas définissant les champs de métadonnées pour XMP, ces schémas pouvant être étendus par l'utilisateur. Dans la mesure du possible, les paquets XMP sont intégrés au sein même des fichiers¹³, ce qui est possible notamment avec les formats JPEG [13], TIFF [28], GIF [8], PNG [23], PDF [19], SVG [26] (basé sur XML) et EPS [25]. Les applications incompatibles XMP peuvent continuer à manipuler des fichiers intégrant des paquets XMP ; seules les applications XMP peuvent lire ou écrire les métadonnées.

¹³Dans le cas contraire, un fichier externe doit accueillir les métadonnées.

6.3 Métadonnées noyées dans l'image : filigranage et stéganographie

Le filigranage et la stéganographie sont des techniques dégradatrices permettant d'ajouter des données de nature quelconque au sein d'une image. Elles peuvent notamment être utilisés afin d'intégrer des métadonnées relatives à l'image en faisant abstraction du format utilisé.

6.3.1 Filigranage

Le filigranage (*watermarking*) a pour objectif, par exemple, de rajouter des données de gestion de droits (auteur, licence de l'image) facilement lisibles mais résistants à des procédés de transformation usuels comme le changement de format avec compression destructive, les transformations géométriques élémentaires (rognage, homothétie, ...), l'impression ou la numérisation par scanner. La résistance aux transformations va de pair avec une dégradation de la qualité d'image et la visibilité du filigranage.

6.3.2 Stéganographie

La stéganographie se définit par l'ajout de données secrètes camouflées au sein d'autres données, ici une image numérique. L'objectif est de pouvoir communiquer des données confidentielles au sein d'images anodines sans éveiller l'attention. Contrairement au filigranage, la résistance aux modifications de l'image n'est pas recherchée mais plutôt la furtivité de l'information insérée, ce qui interdit l'usage de champs usuels de métadonnées. Généralement les données sont préalablement chiffrées avec une clé secrète ou un algorithme de chiffrement asymétrique avant d'être camouflées dans l'image. Une technique élémentaire de camouflage au sein d'un fichier de format brut consiste à remplacer les bits de poids faibles des composantes chromatiques de l'image par les données à camoufler¹⁴.

7 Images bitmap dans les formats composites

7.1 Formats natifs d'applications de manipulation d'images bitmap

Les applications de retouche d'image proposent des formats de travail intégrant l'image bitmap retouchée avec les calques, les sélections ainsi que d'autres informations de travail. Généralement ces formats sont peu documentés et uniquement destinés au travail de retouche sur une image (et non pour l'échange de fichiers). Parmi ces formats natifs, on peut citer :

- Le format XCF de Gimp. Ce format n'intègre pas de compression native, il est donc généralement utilisé en conjonction avec une application de compression généraliste telle que gzip ou bzip2. Il est censé offrir une compatibilité ascendante.
- Le format PSD de Photoshop.

7.2 Formats d'images à palette animées bitmap

Certains formats d'images bitmap à couleurs indexées offrent la possibilité de réaliser des images animées en conservant plusieurs images, voire leur différences successives. Le plus couramment utilisé sur le Web est le format GIF ; une extension du PNG existe également (MNG : *Multiple Network Graphics* [16]) mais s'avère moins populaire.

7.3 Formats vectoriels de description de page

Les formats de description de page sont utilisés afin de décrire de façon non-ambigüe le contenu d'une page pour garantir un rendu identique sur un écran ou sur du papier après impression.

¹⁴La visibilité à l'oeil nu est généralement impossible mais des techniques statistiques permettent de déceler une distribution anormalement aléatoire des bits de poids faible.

7.3.1 Le format PostScript

PostScript [25] est un langage de description de page Turing-complet¹⁵ développé par Adobe (la spécification est librement implantable¹⁶) pour être implanté à l'origine dans les imprimantes. Il dispose de commandes permettant de réaliser des opérations arithmétiques, des tracés de segments et courbes de Bézier avec application possible de matrices de transformations géométriques (composition de translation, rotations, et homothéties). Il dispose également d'un système de gestions de fontes et permet l'insertion de textes. Certaines commandes permettent de dessiner des matrices de pixels (images bitmap) monochromes, en niveaux de gris voire en couleurs avec une base RVB, CMJN ou TSV.

Le format Encapsulated PostScript (EPS) Cette variante de PostScript introduit des indications supplémentaires qui rendent l'intégration de fichiers EPS plus simple au sein de documents PostScript : il s'agit principalement de la spécification de la zone frontière de l'image (*BoundingBox*) ainsi que d'une pré-visualisation de basse qualité au format PICT, TIFF ou WMF¹⁷.

7.3.2 Le format PDF

Le format PDF (*Portable Document Format*) [19] est une évolution de PostScript proposée par Adobe en 1993 destinée aussi bien à l'affichage à l'écran qu'à l'impression de documents avec rendu uniforme. Contrairement à PostScript, le rendu de documents PDF ne nécessite qu'un interpréteur simple, le jeu d'instructions PDF étant plus limité¹⁸. Tout comme PostScript, PDF propose des commandes pour le dessin vectoriel et peut abriter des fontes. Les versions les plus récentes du format PDF permettent (avec les visualiseurs compatibles) d'intégrer des éléments multimédias tels que des images bitmap compressées, des vidéos, des liens hypertextes, des scripts JavaScript ainsi que des métadonnées telle qu'une table de matière, un index ou des notes. La version 1.4 apporte un système de gestion de droits (avec chiffrement symétrique AES [1] 128 bits). Le support des calques est également présent à partir de la version 1.5. Concernant les images bitmap, le format PDF supporte des images sans compression destructive avec diverses bases (RGB, CMJN et TSV avec support ICC) ainsi que des images au format JPEG [13] ou JPEG2000 [14]. La version 1.6 introduit le support d'images vectorielles 3D.

L'implantation logicielle du format PDF est libre ce qui a permis la création de nombreux générateurs et visualiseurs libres¹⁹. Cependant seuls les logiciels de visualisation et génération de Adobe (Reader et Acrobat) permettent l'exploitation de toutes les fonctionnalités des dernières versions.

Il existe des sous-spécifications du format PDF faisant l'objet de standardisation par l'organisation internationale de normalisation : PDF/A [20] (ISO 19005) pour l'archivage de documents basé sur PDF 1.4 et PDF/X-3 [21] (ISO 15930) pour l'utilisation en pré-pressé basé sur PDF 1.3.

7.3.3 Autres formats de description de pages

Les formats PostScript et PDF sont très majoritairement utilisés pour la description de page, nous citons quelques autres formats moins utilisés :

¹⁵En théorie, n'importe quel programme informatique peut être développé en langage PostScript (bien que cela ne soit pas sa vocation) comme un serveur HTTP [37] ou un jeu de la vie [38].

¹⁶Des logiciels libres de génération et de visualisation de fichiers PostScript existent tels que GhostScript [32] et GhostView.

¹⁷Cette fonctionnalité était utilisée anciennement pour la visualisation rapide du fichier EPS pour les machines de faible puissance. Elle n'est pratiquement plus utilisée actuellement où la majorité des programmes interprètent directement les commandes PostScript pour la visualisation à l'écran.

¹⁸Le langage PDF ne comprend pas par exemple des instructions conditionnelles ou de boucles et n'est pas Turing-complet. Seules les instructions utiles pour le rendu graphique sont conservées.

¹⁹Comme *GhostScript*, *XPDF*, *KPDF*, ...

- Le format PICT [22] : ce format initialement utilisé sur les systèmes d'exploitation Macintosh permet de combiner texte, dessin vectoriel ou matrices de pixels. Son utilisation est actuellement marginale.
- Le format DVI (*DeVice Independent*) [6] : ce format de description de page est utilisé par \TeX . Le fichier DVI n'abrite ni images ni fontes ce qui rend son intérêt limité pour l'échange de fichiers.
- Le format XPS (*XML Paper Specification*) [30] — surnommé Metro — : ce format proposé basé sur XML proposé par Microsoft a pour ambition de concurrencer le PDF.
- Le format SVG (*Scalable Vector Graphics*) [26] : ce format standardisé par le W3C est utilisé pour l'affichage de figures graphiques vectorielles fixes ou animées. Il permet d'incorporer du texte, des liens hypertextes, du son ainsi également que des images bitmap au format JPEG ou PNG (encapsulation ou liens vers des images externes). Le format SVG est en fait un format XML (la spécification prévoit l'éventuelle compression avec *gzip*).

8 Conclusion

Nous avons abordé par une démarche transversale les principaux formats utilisés pour les images bitmap. On notera que certaines techniques de compression évoquées dans ce document (telle que la compression fractale) ne font malheureusement pas l'objet de formats couramment utilisés. De même, on pourra regretter que certains formats (tel que JPEG2000) possédant des qualités manifestes ne fasse pas l'objet d'une plus large diffusion. Il convient également de souligner l'importance de la mise au point de formats libres afin de faciliter leur adoption aussi bien par les industriels que par les développeurs de logiciels commerciaux et libres : l'existence de brevets logiciels connus ou inconnus pose malheureusement une incertitude sur certains formats²⁰.

²⁰Par exemple l'existence de brevets sur des techniques de compression par ondelettes pourrait rendre problématique l'utilisation libre du format JPEG2000. Il existe cependant des bibliothèques libres de codage/décodage JPEG2000 : OpenJPEG [36] et JasPer [34].

References

- [1] Advanced Encryption Standard. URL <http://csrc.nist.gov/CryptoToolkit/aes/>.
- [2] Adobe RGB color image encoding, 1998. URL <http://www.adobe.com/digitalimag/adobergb.html>.
- [3] Windows bitmap format. URL http://msdn.microsoft.com/library/default.asp?url=/library/en-us/gdi/bitmaps_4v1h.asp.
- [4] Commission Internationale de l'Éclairage. URL <http://www.cie.co.at/cie/>.
- [5] Digital Negative Specification v 1.1.0.0. URL http://www.adobe.com/products/dng/pdfs/dng_spec.pdf.
- [6] Description of the DVI file format. URL <http://www.math.umd.edu/~asnowden/comp-cont/dvi.html>.
- [7] Flashpix format. URL http://www.i3a.org/i_flashpix.html.
- [8] GIF89a specification. URL <http://www.w3.org/Graphics/GIF/spec-gif89a.txt>.
- [9] Internet Assigned Numbers Authority. URL <http://www.iana.org/>.
- [10] International Color Consortium. URL <http://www.color.org/>.
- [11] IPTC metadata for XMP. URL <http://www.iptc.org/IPTC4XMP/>.
- [12] JPEG file interchange format (JFIF). URL <http://www.w3.org/Graphics/JPEG/jfif3.pdf>.
- [13] ITU t.81 JPEG compression, . URL <http://www.w3.org/Graphics/JPEG/itu-t81.pdf>.
- [14] JPEG2000, . URL <http://www.jpeg.org/jpeg2000/>.
- [15] MIME image media types. URL <http://www.iana.org/assignments/media-types/image/>.
- [16] MNG homepage. URL <http://www.libpng.org/pub/mng/>.
- [17] Normes MPEG. URL <http://www.iso.org/iso/fr/prods-services/popstds/mpeg.html>.
- [18] Zsoft PCX file format technical reference manual. URL <http://www.qzx.com/pc-gpe/pcx.txt>.
- [19] PDF Reference, fifth edition, version 1.6, . URL <http://partners.adobe.com/public/developer/en/pdf/PDFReference16.pdf>.
- [20] Papier-blanc sur le pdf/a, . URL <http://www.pdf-tools.com/public/downloads/whitepapers/whitepaper-pdfa.pdf>.
- [21] pdfx.info (site d'information sur le pdf/x), . URL <http://www.pdfx.info/>.
- [22] PICT format. URL http://developer.apple.com/technotes/qd/qd_14.html.
- [23] Portable Network Graphics (png). URL <http://www.w3.org/Graphics/PNG/>.
- [24] NetPNM : implementation for portable pixmap formats (PNM). URL <http://netpbm.sourceforge.net/doc/>.
- [25] PostScript language reference, third edition. URL <http://partners.adobe.com/public/developer/en/ps/PLRM.pdf>.
- [26] SVG format specification. URL <http://www.w3.org/TR/SVG/>.

- [27] Truevision tga specification 2.0. URL <http://www.dca.fee.unicamp.br/~martino/disciplinas/ea978/tgaffs.pdf>.
- [28] Tagged Image File Format (tiff) revision 6.0. URL <ftp://ftp.sgi.com/graphics/tiff/TIFF6.ps>.
- [29] WBMP format. URL <http://www.wapforum.org/what/technical/SPEC-WAESpec-19990524.pdf>.
- [30] XML Paper Specification. URL <http://www.microsoft.com/whdc/xps/xpsspec.msp>.
- [31] Logiciel de compression bzip2. URL <http://www.bzip.org/>.
- [32] Ghostscript. URL <http://www.ghostscript.com/>.
- [33] The gzip Home Page. URL <http://www.gzip.org/>.
- [34] JasPer. URL <http://www.ece.uvic.ca/~mdadams/jasper/>.
- [35] libpng Home Page. URL <http://www.libpng.org/pub/png/libpng.html>.
- [36] OpenJPEG. URL <http://www.openjpeg.org/>.
- [37] PSHTTPD (serveur HTTP programmé en PostScript). URL <http://public.planetmirror.com/pub/pshttpd>.
- [38] Life in PostScript (jeu de la vie programmé en PostScript). URL <http://www.tjhsst.edu/~edanaher/pslife/>.
- [39] Wotsit. URL <http://www.wotsit.org>.
- [40] The zlib Home Site. URL <http://www.zlib.net/>.
- [41] W. Chen, C.H. Smith, and S.C. Fralick. A fast computational algorithm for the discrete cosine transform. *IEEE Trans. on Communications*, COM-25:1004–1009, 1977.
- [42] D.A. Huffman. A method for the construction of minimum-redundancy codes. pages 1098–1102, sept 1952. URL http://compression.ru/download/articles/huff/huffman_1952_minimum-redundancy-codes.pdf.
- [43] A.E. Jacquin. Image coding based on a fractal theory of iterated contractive image transformations. *IEEE Transactions On Image Processing*, 1:18–30, 1992. ISSN 1057-7149.
- [44] Michael Stokes, Matthew Anderson, Srinivasan Chandrasekar, and Ricardo Motta. A standard default color space for the internet - sRGB, 1996. URL <http://www.w3.org/Graphics/Color/sRGB>.
- [45] D. Taubman. High performance scalable image compression with EBCOT. *IEEE Transactions on Image Processing*, 2000. URL <http://www.ece.umn.edu/users/pengkw/papers/EBCOT2000.pdf>.
- [46] T. A. Welch. A technique for high-performance data compression. *Computer*, 17:8–19, 1984.
- [47] Jacob Ziv and Abraham Lempel. A universal algorithm for sequential data compression. *IEEE Transactions on Information Theory*, 1977.
- [48] Jacob Ziv and Abraham Lempel. Compression of individual sequences via variable-rate coding. *IEEE Transactions on Information Theory*, 1978.

Index

BMP, 3, 7

DNG, 4

DVI, 16

EPS, 13

FIF, 13

FPX, 5

GIF, 4, 5, 7, 13

JFIF, 10

JPEG, 1, 3, 5, 6, 9, 13, 15

JPEG2000, 6, 11, 15

Metro, 16

MNG, 14

PBM, 2

PCX, 7

PDF, 13, 15

PDF-A, 15

PDF-X, 15

PICT, 15, 16

PNG, 1, 4, 5, 7, 13

PNM, 2

PPM, 2

PS, 15

Raster, 3

SVG, 13, 16

TGA, 3

TIFF, 2–5, 7, 13, 15

WBMP, 2

WMF, 15

XPS, 16