



Chapitre 5

LA COMPRESSION DES IMAGES NUMÉRIQUES

PROBLÈMES TRANSMISSION / STOCKAGE

× Problèmes :

- + Non compressée (c'est-à-dire sous forme d'un fichier BMP par exemple), une image couleur (en RVB) haute résolution, en 1600 x 1200, occupe 5.5 Mo.
- + Ceci ne permet pas :
 - × une transmission efficace de l'image : sur un réseau à 100 Mbits/s, la transmission prendrait 0.33 s.
 - × une sauvegarde efficace de l'image : sur une carte mémoire de 1 Go, on ne pourrait sauvegarder que 186 images.

INTÉRÊTS DE LA COMPRESSION

× Intérêt :

- + Gagner de la place
- + Diminuer le temps de transmission

× Deux types de compression :

1. Sans perte d'information :

le signal décompressé est identique au signal non compressé :
RLE, Huffman, LZW, Arithmétique...

2. Avec perte d'information :

on sacrifie des informations pour compresser plus efficacement
: *DCT, ondelettes, prédictif*

COMPRESSION SANS PERTE : RLE

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

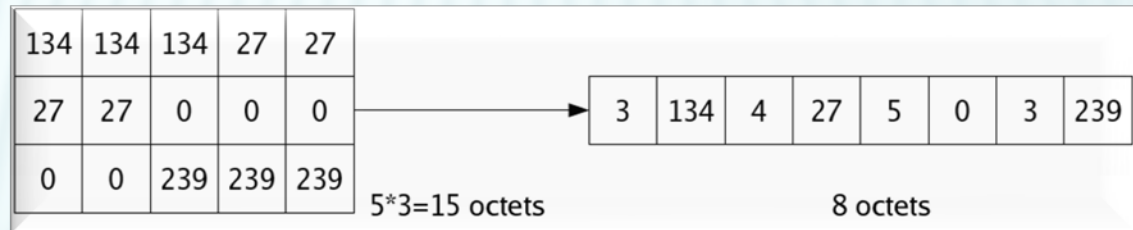
- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Définition (*Run-Length Encoding*)

- + Algorithme de compression qui consiste à indiquer pour chaque suite de données d'une même valeur, le nombre de d'occurrences correspondantes.

× Exemple sur une image niveaux de gris



× Problème

- + Inefficace sur des images très texturées

COMPRESSION SANS PERTE : RLE

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Exemples de formats d'image

+ BMP : *Windows* et *OS/2*

× Fichier en 3 zones :

- ★ en-tête, palette (optionnelle), image

× Codage des pixels en 1, 4, 8, 16 ou 24 bits

× Compression RLE :

- ★ pour les images 8 et 4 bits

× Remarques :

- ★ quasiment inexistant sur le Web
- ★ excellente qualité (aucune dégradation d'image)
- ★ très lourd

COMPRESSION SANS PERTE : RLE

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Exemples de formats d'image

+ PCX : société ZSoft Corporation

× Fichier en 4 zones :

- ✱ en-tête, image, information, palette (optionnelle)

× Nombre de couleurs :

- ✱ 2 couleurs (1 bit), 16 (4 bits), 256 (8 bits) ou 16,8 millions (24 bits)
- ✱ en 24 bits/pixel : découpage en 3 plans de couleur

COMPRESSION SANS PERTE : RLE

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Exemples de formats d'image

+ PCX : société ZSoft Corporation

× Compression RLE :

★ mettre les couleurs :

- × les plus fréquentes aux premières positions (0x00 ... 0xC0)
- × les moins fréquentes aux dernières positions (0xC1 ... 0xFF) > compression RLE

× Remarques :

- ★ compression rapide, peu exigeant en mémoire
- ★ pas très efficace, surtout pour les images issues du monde réel

COMPRESSION SANS PERTE : LZW

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Définition (Lempel-Ziv-Welch)

- + Algorithme de compression qui construit dynamiquement un dictionnaire à partir des données à compresser, pour remplacer les chaînes de caractères du dictionnaire par un nouveau code.

× Dictionnaire

+ Construction

- × Initialisé avec les 256 valeurs de la table ASCII.
- × Découpage des données en chaînes d'octets, comparées au dictionnaire et ajoutées si jamais elle n'y est pas présente.

+ Compression

- × Parcourt des données en les codant ; si jamais une chaîne est plus petite que le plus grand mot du dictionnaire, elle est transmise.

+ Décompression

- × Le dictionnaire est reconstruit dans le sens inverse, et n'a donc pas besoin d'être stocké.

COMPRESSION SANS PERTE : LZW

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Exemples de formats d'image

+ TIFF : Adobe

- × Stockage d'images BITMAP de taille importante (plus de 4 Go compressées)

× Nombre de couleurs :

- ★ 2 couleurs (1 bit), 16 (4 bits), 256 (8 bits), 65 536 (16 bits), 16,8 millions (24 bits) ou 4 milliards (32 bits)

× Espaces de couleurs :

- ★ Non limité au RVB
- ★ Possibilité d'utiliser CMJN, YCbCr, ...

COMPRESSION SANS PERTE : LZW

compression

× **sans perte**

- + RLE
- + **LZW**
- + Huffman
- + arithmétique

× **avec pertes**

- + DCT
- + ondelettes
- + prédictif

× **comparaisons**

× Exemples de formats d'image

+ **TIFF : Adobe**

× Structure du fichier TIFF

- ✱ Utilisation de balises décrivant les caractéristiques de l'image (dimensions de l'image, nombre de couleurs utilisées, type de compression, correction gamma, etc.)

× Algorithmes de compression

- ✱ *RLE / JPEG / LZW / ...*

× *Remarques :*

- ✱ Possibilités qu'un logiciel ne puisse pas lire le format TIFF créé sur un autre logiciel

COMPRESSION SANS PERTE : LZW

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Exemples de formats d'image

+ GIF : *CompuServe*

× Caractéristiques

- ✧ compression LZW (propriétaire)
- ✧ entrelacement pour affichage progressif
- ✧ limité à 256 couleurs parmi 16.8 millions dans sa palette
- ✧ permettre les images animées
- ✧ couleur transparente

× Remarques

- ✧ inefficace pour les photos réalistes
- ✧ efficace pour les logos et graphiques

COMPRESSION SANS PERTE : HUFFMAN

compression

× **sans perte**

- + RLE
- + LZW
- + **Huffman**
- + arithmétique

× **avec pertes**

- + DCT
- + ondelettes
- + prédictif

× **comparaisons**

× Définition (*Huffman*)

- + Algorithme de compression de type statistique, basé sur l'entropie de Shannon.
- + Grâce à une méthode d'arbre, il permet de coder les octets revenant le plus fréquemment avec une séquence de bits beaucoup plus courte.

× Remarque

- + Cet algorithme offre des taux de compression démontrés les meilleurs possibles pour un codage par **symbole**.

COMPRESSION SANS PERTE : HUFFMAN

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× 3 variantes de l'algorithme de Huffman :

+ statique :

- × Chaque octet a un code prédéfini par le logiciel.
- × Compression seulement sur un seul type de fichier

+ semi-adaptatif :

- × Le fichier est d'abord lu, de manière à construire un arbre utile pour la compression / décompression
- × Nécessité de transmettre l'arbre pour la décompression

+ adaptatif :

- × Compression toujours optimale car l'arbre est construit dynamiquement au fur et à mesure de la compression du flux
- × Temps d'exécution plus long
- × Inutile de transmettre ou stocker l'arbre

COMPRESSION SANS PERTE : HUFFMAN

compression

× **sans perte**

- + RLE
- + LZW
- + **Huffman**
- + arithmétique

× **avec pertes**

- + DCT
- + ondelettes
- + prédictif

× **comparaisons**

× Remarque

- + Quasi tous les compresseurs l'utilisent comme 2nd étage de compression, recomprimant ce qui l'a été par une autre technique.

× Exemples de format d'archive

+ Le format **GZIP**

- × Format *deflate* basé sur la compression sans perte combinée : RLE, LZ77 (algorithme précurseur du LZW), Huffman

+ Le format **BZIP2**

- × Utilise la transformée de Burrows-Wheeler avec le codage de Huffman. Le taux de compression est meilleur que celui du format GZIP.

COMPRESSION SANS PERTE : HUFFMAN

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Exemples de formats d'image

+ PNG : *Unisys*

× Alternative libre au format GIF

× Structure

- ★ En-tête
- ★ Découpage de l'image en segment contenant :
 - × Taille et nature du segment, les données compressées, code correcteur CRC

× Algorithmes de compression

- ★ Deflate, basé sur le format ZIP (combinaison de la compression LZ77 et Huffman)

COMPRESSION SANS PERTE : HUFFMAN

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Exemples de formats d'image

+ PNG : *Unisys*

× Remarques

- ★ Libre
- ★ 8 ou 16 bits / composante de couleurs
- ★ Jusqu'à 24 ou 48 bits / pixel
- ★ Images binaires, couleurs réelles ou indexées
- ★ Transparence par couche alpha : 256 niveaux de transparence possible (contre 1 pour le GIF)
- ★ Bonne compression mais très vite limitée comparée aux méthodes de compression avec pertes

COMPRESSION SANS PERTE : HUFFMAN

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Exemples de formats d'image

+ JPEG : *Joint Photographic Experts Group*

- × Méthode de compression d'image basée sur la transformée DCT en 1^{er} étage et la compression sans perte de Huffman dans le 2nd étage.
- × De plus amples explications dans la section « compression avec pertes : DCT »

COMPRESSION SANS PERTE : ARITHM.

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Définition

- + Algorithme de compression qui s'apparente au codage de Huffman, sauf qu'il encode le message entièrement et le représente par un seul nombre n (flottant) alors que les autres codages séparent le message d'entrée en les symboles qui le composent et encodent ensuite chaque symbole par un mot code.

× Remarque

- + Cet algorithme offre des taux de compression bien meilleurs que le codage de Huffman.
- + Aucun format d'images l'utilise : très lent

COMPRESSION AVEC PERTE : DCT

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Définition

- + algorithme de compression basé sur la **transformée DCT** (*Discrete Cosine Transform*, en français *transformée en cosinus discrète*), variante de la *transformée de Fourier*.
- + Cette méthode permet de décrire chaque bloc en une carte de fréquences et en amplitudes plutôt qu'en pixels et couleurs.

× Remarques

- + Fréquence → importance et rapidité d'un changement de couleur
- + Amplitude → écart pour chaque changement de couleur
- + Utilisée pour les formats JPEG et MPEG.

COMPRESSION AVEC PERTE : DCT

compression

× sans perte

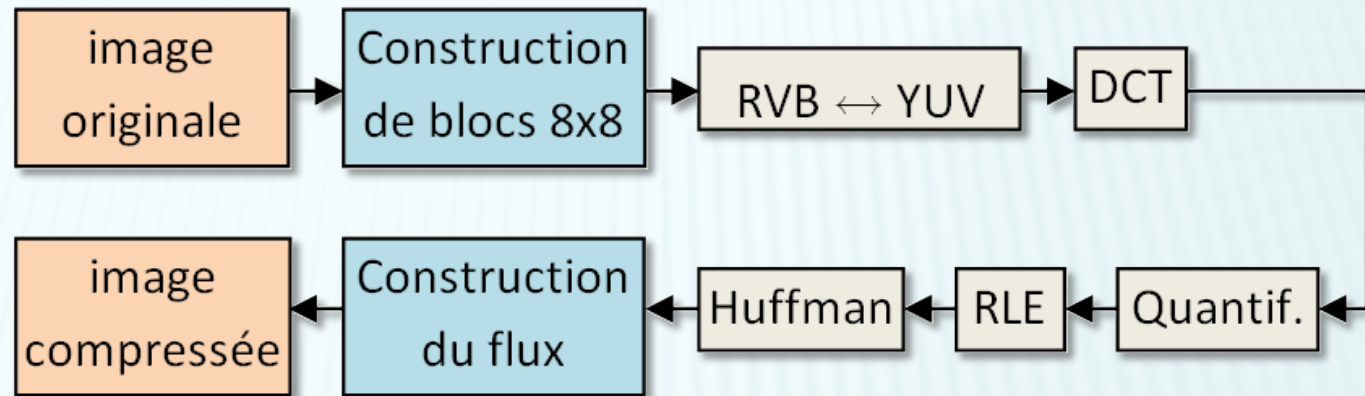
- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Chaîne de traitement du format JPEG



× Remarques

- + Chaque image est décomposée en blocs 8x8
- + Sur les bords, on peut compléter les derniers blocs
 - × Par des zéros
 - × Par des prolongements continus
 - × Par une symétrie
- + Chaque bloc est une 'petite' image

COMPRESSION AVEC PERTE : DCT

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

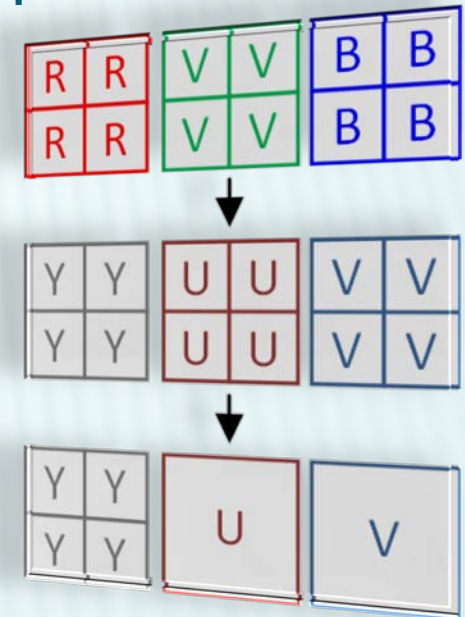
× Conversion RVB $\leftrightarrow$ YUV (Y : luminance, U et V : chrominances)

+ Système de conversion réversible :

- × $Y = 0,299 R + 0,587 V + 0,114 B$
- × $U = -0,1687 R - 0,3313 V + 0,5 B$
- × $V = 0,5 R - 0,4187 V - 0,0813 B$

+ L'œil est plus sensible à l'intensité lumineuse qu'aux informations colorées.

- × A cette étape, on réduit le nombre de chrominance par 4.
- × Ainsi, le nombre de valeurs est réduit par 2.



+ A cette étape, l'image a pu être dégradée de 2 façons :

- × perte de précision lors de la conversion RGB $\rightarrow$ YUV entre les entiers et les flottants.
- × diminution de la précision du codage de la chrominance.

COMPRESSION AVEC PERTE : DCT

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Transformée en cosinus discret – DCT

- + Calcul de la DCT sur un sous-bloc 8x8

$$DCT(i, j) = \frac{1}{\sqrt{2}} C(i) C(j) \sum_{x=0}^7 \sum_{y=0}^7 \left[pixel(x, y) \cos\left(\frac{(2x+1)i\pi}{16}\right) \cos\left(\frac{(2y+1)j\pi}{16}\right) \right]$$

$$\begin{cases} C(x) = \frac{1}{\sqrt{2}}, \text{ si } x = 0 \\ C(x) = 1, \text{ si } x > 0 \end{cases}$$

× Remarques

- + Chaque sous-bloc en 8x8 est transformée en une autre matrice par la DCT
- + Il n'y a pas de perte d'information ici

								1.0e+003 *								
144	189	188	188	201	203	198	170	DCT →	1.4195	-0.0078	-0.0140	-0.0113	-0.0507	-0.0280	-0.0359	0.0002
159	196	182	173	185	195	205	188		0.0404	-0.0401	-0.0302	0.0259	-0.0183	0.0003	-0.0001	-0.0001
153	191	184	182	196	202	206	187		0.0402	-0.0183	-0.0109	-0.0342	0.0004	-0.0000	-0.0000	0.0004
170	201	187	186	199	199	194	170		-0.0301	0.0125	0.0005	0.0199	-0.0002	0.0001	-0.0001	-0.0001
158	170	135	120	130	132	135	118		-0.0647	0.0300	0.0001	-0.0004	0.0005	-0.0001	0.0002	-0.0000
180	199	174	167	176	176	182	171		0.0848	-0.0250	-0.0390	-0.0001	0.0000	0.0000	0.0002	-0.0004
162	200	197	196	196	186	196	195		0.0004	0.0005	-0.0000	0.0002	0.0004	0.0003	0.0004	-0.0001
159	196	186	168	150	136	161	178		-0.0501	0.0001	-0.0002	-0.0004	0.0001	0.0001	0.0004	0.0006

COMPRESSION AVEC PERTE : DCT

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

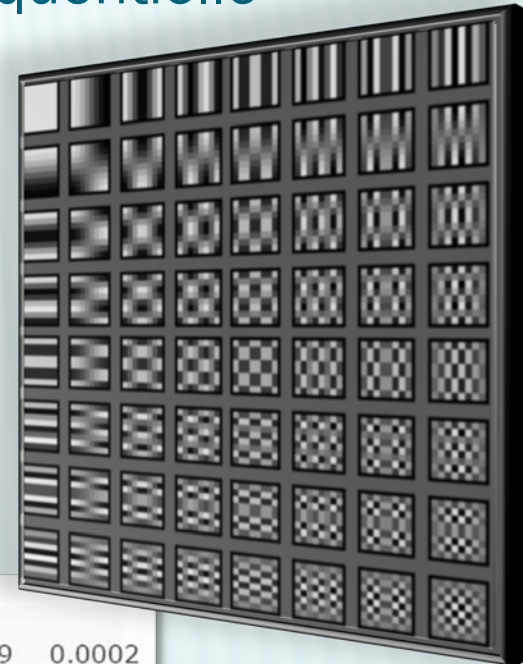
× Transformée en cosinus discret – DCT

+ C'est une représentation fréquentielle

× Composante continue
au coin en haut à gauche

× Basses fréquences
en haut à gauche

× Hautes fréquences
en bas à droite



1.0e+003 *

1.4195	-0.0078	-0.0140	-0.0113	-0.0507	-0.0280	-0.0359	0.0002
0.0404	-0.0401	-0.0302	0.0259	-0.0183	0.0003	-0.0001	-0.0001
0.0402	-0.0183	-0.0109	-0.0342	0.0004	-0.0000	-0.0000	0.0004
-0.0301	0.0125	0.0005	0.0199	-0.0002	0.0001	-0.0001	-0.0001
-0.0647	0.0300	0.0001	-0.0004	0.0005	-0.0001	0.0002	-0.0000
0.0848	-0.0250	-0.0390	-0.0001	0.0000	0.0000	0.0002	-0.0004
0.0004	0.0005	-0.0000	0.0002	0.0004	0.0003	0.0004	-0.0001
-0.0501	0.0001	-0.0002	-0.0004	0.0001	0.0001	0.0004	0.0006

COMPRESSION AVEC PERTE : DCT

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Quantification

- + Processus principalement responsable de la dégradation de l'image.
- + Grâce à un facteur de qualité, on crée une table de quantification 8x8 selon la formule :

$$quantification[i][j] = 1 + (i + j + 1) \times \text{qualité}$$

- + Exemple pour un facteur de qualité de 3 :

$$quantification = \begin{bmatrix} 4 & 7 & 10 & 13 & 16 & 19 & 22 & 25 \\ 7 & 10 & 13 & 16 & 19 & 22 & 25 & 28 \\ 10 & 13 & 16 & 19 & 22 & 25 & 28 & 31 \\ 13 & 16 & 19 & 22 & 25 & 28 & 31 & 34 \\ 16 & 19 & 22 & 25 & 28 & 31 & 34 & 37 \\ 19 & 22 & 25 & 28 & 31 & 34 & 37 & 40 \\ 22 & 25 & 28 & 31 & 34 & 37 & 40 & 43 \\ 25 & 28 & 31 & 34 & 37 & 40 & 43 & 46 \end{bmatrix}$$

COMPRESSION AVEC PERTE : DCT

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

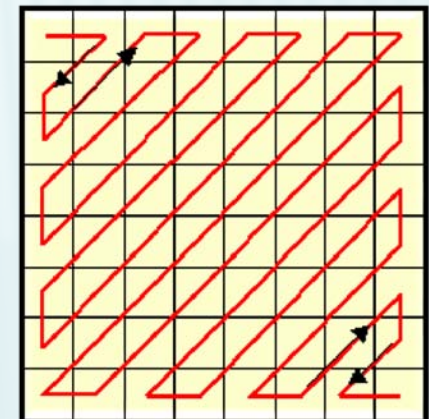
× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Quantification

- + Ensuite chaque coefficient calculé par la DCT va être quantifié afin de mettre à zéro une grande partie des hautes fréquences.
- + La lecture des coefficients se fait en zigzag : on ne garde que la composante continue et quelques basses, voire moyennes fréquences.
- + Les méthodes de compression RLE et Huffman sans perte deviennent ici plus efficaces.



COMPRESSION AVEC PERTE : ONDELETTES

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Définition

- + algorithme de compression basé sur la **transformée en ondelettes** (*Wavelet Transform*), variante de la **transformée de Fourier**. Cette méthode est basée sur le regroupement **multirésolution** de l'information.

× Remarques

- + Séparer de l'information général (basses fréquences) et les détails (hautes fréquences), contenue dans une image, un son, ...
- + Utilisée pour les formats JPEG2000 et MPEG2000.

COMPRESSION AVEC PERTE : ONDELETTES

compression

× sans perte

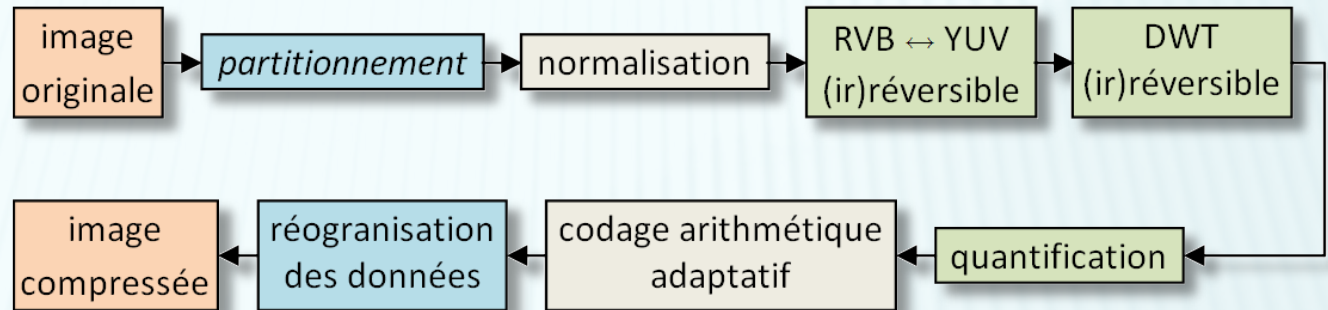
- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Chaîne de traitement du JPEG2000



× Remarques

+ JPEG2000 meilleur que JPEG

- × Grande flexibilité
- × Multi-plate-forme
- × Peu de détériorations de l'image originale
- × Meilleur taux de compression pour une même qualité
- × Meilleure adaptation à des systèmes à faible débit
 - ★ Envoi en priorité de certains bits correspondant à des régions importantes de l'image, qui sera intelligible plus rapidement.
- × Possibilité de compresser avec ou sans perte.

COMPRESSION AVEC PERTE : ONDELETTES

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Partitionnement de l'image

+ Définir des zones (ROI : *Regions Of Interest*)

- × où la qualité doit être préservée
- × mieux encodées, et décodées plus rapidement, pour permettre un affichage optimal par exemple lors de communications réseau.



Avec ROI



Sans ROI

COMPRESSION AVEC PERTE : ONDELETTES

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Phase de normalisation

+ But : meilleurs taux de compression

+ Principe :

× Utilisation d'un « *level offset* »

★ les pixels codés entre 0 et 255,
sont codés entre -128 et 127

× Pour une transformation irréversible (avec perte)

★ Normalisation des pixels entre $-\frac{1}{2}$ et $\frac{1}{2}$

+ Intérêt de la normalisation

× La compression ne dépend plus du nombre de bits nécessaires pour coder l'image.

× JPEG2000 gagne en flexibilité : on pourra coder n'importe quelle profondeur de bit.

COMPRESSION AVEC PERTE : ONDELETTES

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Conversion RVB $\leftrightarrow$ YUV (Y : luminance, U et V : chrominances)

+ Transformation réversible (RCT) :

- × $Y = 0,299 R + 0,587 V + 0,114 B$
- × $U = -0,1687 R - 0,3313 V + 0,5 B$
- × $V = 0,5 R - 0,4187 V - 0,0813 B$

+ Transformation irréversible (ICT) :

- × $Y = \text{partie entière } ((R + 2 V + B) / 4)$
- × $U = R - V$
- × $V = B - V$

COMPRESSION AVEC PERTE : ONDELETTES

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Transformée en ondelettes discrètes 1D

+ Il existe un nombre illimité d'ondelettes

+ Exemple : ondelettes discrètes 1D de Haar

Niveau de détail #	Coefficients de moyenne	Coefficients de détail
0	[9 ; 7 ; 3 ; 5]	
1	$L1 = [8 = (9+7)/2 ; 4 = (3+5)/2]$	$H1 = [1 = (9-7)/2 ; -1 = (3-5)/2]$
2	$L2 = [6 = (8+4)/2]$	$H2 = [2 = (8-4)/2]$

× L : filtre passe-bas (*Low pass filter*)

× H : filtre passe-haut (*High pass filter*)

× Séquence originale : [9 ; 7 ; 3 ; 5]

× Séquence finale : [6 ; 2 ; 1 ; -1]

COMPRESSION AVEC PERTE : ONDELETTES

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Transformée en ondelettes discrètes 2D

- + Construction par succession d'ondelettes 1D discrètes suivant les axes x, puis y de l'image 2D



COMPRESSION AVEC PERTE : ONDELETTES

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Transformée en ondelettes discrètes 2D

+ Exemple : image 2D

*4 niveaux de détail
de la décomposition
en ondelettes 2D*

*Pour chaque niveau
de détail, l'information
générale est en haut à
gauche, et les détails
dans la zone restante*

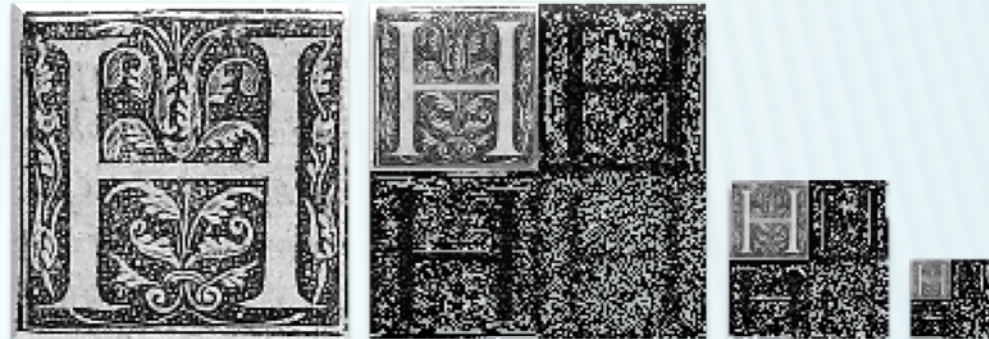
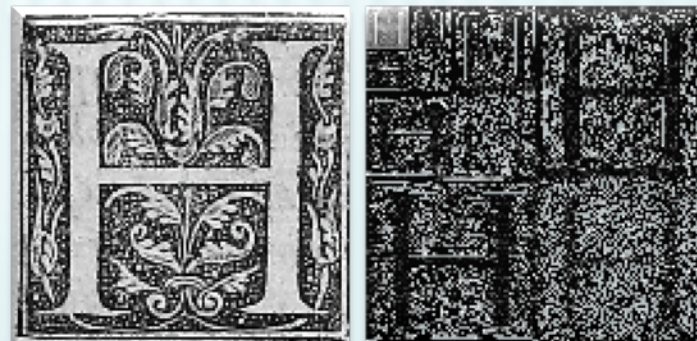


Image initiale à gauche

*Image décomposée avec
les ondelettes 2D à droite*



COMPRESSION AVEC PERTE : ONDELETTES

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

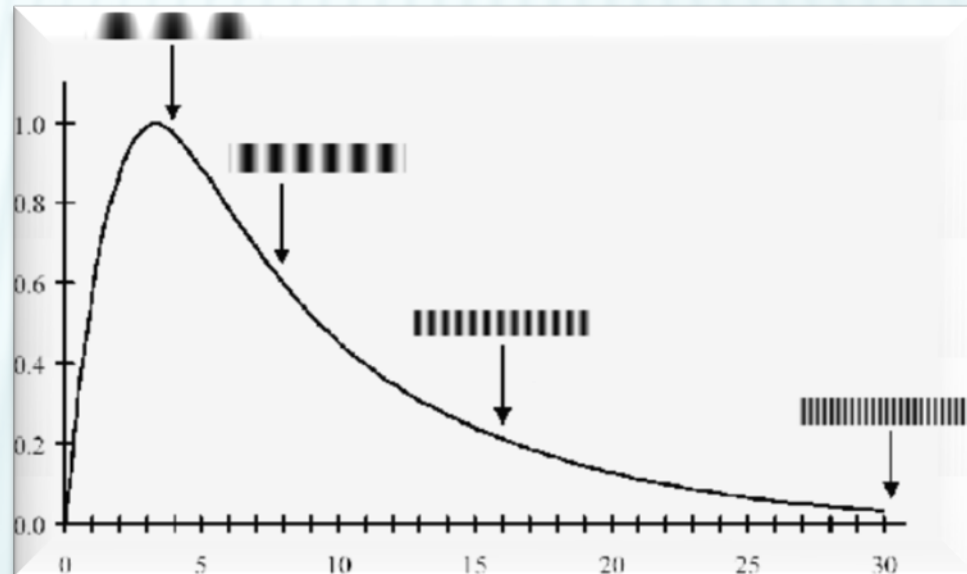
× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Quantification

- + Utilisation d'un quantificateur à *pas variable*
- + Prise en compte du système de vision humaine
 - × Utilisation de la courbe de sensibilité au contraste de l'œil humain en fonction de la fréquence de changement de luminance



COMPRESSION AVEC PERTE : ONDELETTES

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Codage arithmétique adaptatif

- + Utilisation d'un codeur arithmétique adaptatif de type MQ (développé par Mitsubishi).

+ Intérêt du JPEG2000

- × Chaque ROI et chaque niveau de détail est codé indépendamment.
- × Meilleurs résultats qu'un codage global (tel que le format JPEG)

COMPRESSION AVEC PERTE : ONDELETTES

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Organisation du flux de données

+ Les données sont réorganisées pour proposer une reconstruction permettant d'avoir :

- × Rendu Basse Résolution - Haute Qualité
- × Rendu Haute Résolution - Basse Qualité
- × Organisation maximisant le rapport Signal/Bruit
- × Organisation optimisant le rendu visuel
- × ...

COMPRESSION AVEC PERTE : PREDICTIF

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Qu'est ce que le format WebP ?

- + Format d'image matricielle développé par Google
- + Exploite un algorithme de compression avec pertes prédictif utilisé pour les images clés (intra) du codec vidéo VP8
- + Inclut dans un conteneur léger et extensible RIFF

× Remarque

- + Peut contenir jusqu'à 4 métadonnées supplémentaires : le nom de l'auteur, le titre, les droits d'auteurs et des commentaires
- + Format ouvert et récent
- + Taille maximale d'une image WebP : 268 mégapixels

COMPRESSION AVEC PERTE : PREDICTIF

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Motivations :

+ Réduire la quantité de données circulant sur l'Internet : les images constituent environ 65% des données

+ Tests réalisés par Google :

- × un million d'images **déjà compressées** avec pertes
- × en moyenne le WebP réduirait la taille des fichiers de 39 % par rapport aux formats JPEG, PNG et GIF, sans perte de qualité perceptible.

COMPRESSION AVEC PERTE : PREDICTIF

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Fonctionnement :

1. Découpage en blocs (comme le JPEG)
 - × Profondeur de couleurs : 8 bits
 - × Modèle YUV utilisé : YUV 4:2:0
2. Pour chaque bloc :
 - × Prédiction à partir des valeurs des trois blocs (en haut et à gauche)
 - × Prédiction selon quatre modes distincts :
 - ★ Horizontal, vertical, DC (une couleur) et TrueMotion.
 - × Les données malprédites ou non prédites sont compressées en sous-blocs 4x4 avec une transformée DCT et Walsh-Hadamard.
3. La sortie est compressée avec un codage entropique (Huffman ou arithmétique).

COMPRESSION AVEC PERTE : PREDICTIF

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Critiques par Jason Garrett-Glaser :

- + Développeur du codec vidéo x264
- + Il a fait état que :
 - × entre les codecs JPEG, x264 et WebP, la qualité WebP est la pire des trois à cause d'un estompage de l'image dans la plupart des cas.
- + Raison :
 - × Forte optimisation du rapport signal sur bruit
 - × Sans tenir compte d'une bonne optimisation psychovisuelle.

COMPARAISONS : RÉCAPITULATIF

compression

× sans perte

- + RLE
- + LZW
- + Huffman
- + arithmétique

× avec pertes

- + DCT
- + ondelettes
- + prédictif

× comparaisons

× Sans compression : BMP

- + Format très lourd

× Compression sans perte

- + RLE : PCX
 - × Poids plus faible : élimination des redondances par répétition
- + LZW : GIF / TIFF
 - × Utilisation d'un dictionnaire : plus efficace que RLE

× Compression avec pertes

- + Compression avec DCT : JPEG
 - × Réduction du coût mémoire et taux de compression réglable
- + Compression avec ondelettes : JPEG 2000
 - × Meilleur rapport taux de compression / qualité d'image
 - × Gestion multirésolution de l'image
 - × Compression avec ou sans perte
- + Compression prédictive : WebP
 - × Meilleur taux de compression que JPEG, JPEG2000 et PNG
 - × Estompage général sans optimisation psychovisuelle
 - × Non disponible à l'heure actuelle