

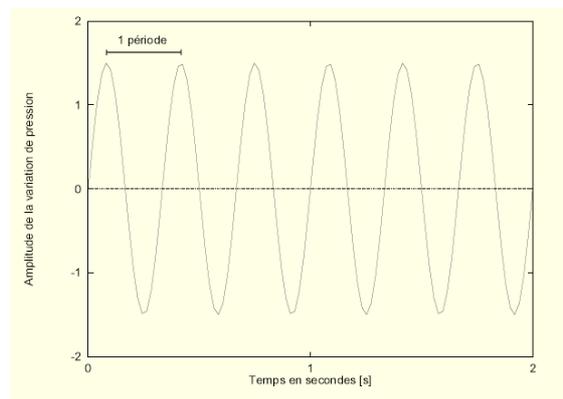
## Chapitre 2 : Les signaux multimédia

### **\*Caractérisation d'un son**

Le son est une onde qui se propage dans l'air et qui est perceptible grâce au détecteur de pression qu'est le tympan, logé dans l'oreille. Ces vibrations sont ensuite converties en signaux transmis au cerveau par le nerf auditif. Le tympan capte ces ondes et les traduites en signaux électriques comme le fait la membrane d'un micro. Un son peut être décrit par quatre paramètres : hauteur tonale, intensité, timbre et durée.

### **Ton ou hauteur tonale**

La hauteur tonale ou ton est liée à la fréquence qui représente le nombre d'oscillations d'un son dans un intervalle de temps donné. Par exemple notre oreille perçoit théoriquement des vibrations acoustiques allant de 15 [Hz] (son grave) à 20000 [Hz] (son aigu).



### **Intensité**

L'intensité est l'amplitude des variations de pression avec lequel notre oreille percevra une note. L'intensité acoustique est la puissance transportée par unité de surface du front d'onde et s'exprime en  $[W/cm^2]$ .

### **Timbre**

Le timbre est ensemble de fréquences et leurs variations d'intensité permettent d'avoir une sorte de carte d'identité du son. Le timbre montre la proportion dans laquelle la fréquence fondamentale du son est mélangée à d'autres fréquences multiples de la fréquence fondamentale et appelées sons harmoniques. Un son qui ne contient aucune harmonique est appelé son pur. Le timbre permet donc d'identifier la source sonore car chaque instrument produit son propre spectre de fréquences.

### **Durée**

La répétition d'une onde sonore donne à son tour la durée du son. Elle s'exprime en secondes et correspond au temps pendant lequel l'énergie du son est perceptible.

### **Analyse en fréquences**

Transformée de FOURIER :

$$X(f) = \int_{-\infty}^{+\infty} x(t)e^{-j2\pi ft} dt$$

Transformée de FOURIER inverse :

$$x(t) = \int_{-\infty}^{+\infty} X(f)e^{j2\pi ft} df$$

Bande passante : L'intervalle de fréquences que peut traiter par un système.

- Bande passante de l'oreille = intervalle de fréquences [15Hz, 20KHz]
- Bande passante du téléphone =[300Hz, 3400Hz]

### **Le format MP3**

Le MP3 est un format propriétaire pour les sons, créé à la fin des années 80. Il se base sur une spécification d'une norme développée par le Moving Pictures Experts Group, intitulée MPEG 1 - Audio Layer 3. MP3 est l'abréviation de ce nom. L'extension de ce format est .mp3.

Le MP3 implémente des méthodes de compression destructrices du son. Pour ce faire, il procède à une transformation en cosinus discrète des échantillons de la séquence à compresser. Ensuite, il code les données avec l'algorithme de Huffman. Toutefois, La norme ne décrit cependant que la méthode de décompression qui doit être utilisée pour les séquences audio de ce format. Dès lors, les développeurs qui désirent concevoir un algorithme conforme à MPEG-1 Audio Layer 3 procèdent à leur guise tant que leur méthode répond aux exigences de la norme.

De plus, le MP3 autorise plusieurs taux de compression. Une seconde de son au format MP3 peut mesurer jusqu'à 128 kilo bits mais il permet aussi d'obtenir des séquences sonores moins compressées. De plus, il autorise une compression à taux constant ou variables. Dans le cas de la compression à un taux constant, les sons sont compressés au même taux tout le long de la séquence. Dans le cas d'une compression à des taux variables, les sons sont compressés à des taux différents, plus ou moins élevés, selon les particularités physiques de la séquence.

### **\*Image**

#### **La lumière**

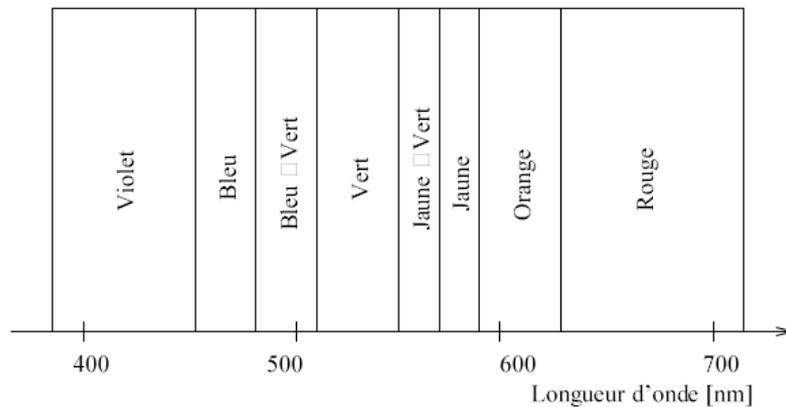
La perception du signal visuel présente des similitudes avec celle du signal audio. De fait, le fond de l'œil est plein de récepteurs qui envoient des influx électriques au cerveau via le nerf optique. De plus,

certaines couleurs ne sont pas perceptibles à cause d'un effet de limitation de la "bande passante" de l'œil.

La lumière couvre une partie du spectre d'énergie électromagnétique. Un rayonnement électromagnétique est en général constitué d'un certain nombre de longueurs d'onde (ou fréquences) que les dispositifs dispersifs de séparer en un spectre. Le spectre est soit discret, soit continu.

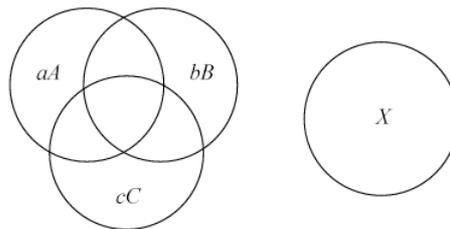
Les longueurs d'onde du spectre visible s'étendent approximativement de 380 à 720 [nm].

Les longueurs d'onde associées aux couleurs :



### L'espace de couleurs additif RGB :

Ce système est fondé sur les trois couleurs monochromatiques, Rouge (700 nm), Vert (546.1 nm) et bleu (435.8 nm).



L'égalisation d'une couleur x au moyen de trois couleurs primaires RGB est défini avec le mélange additif  $aR+bG+cB$ .

### Le système YCbCr :

Où Y, le signal de luminance (noir et blanc), plus deux informations de [chrominance](#), Cb et Cr.

On peut calculer les valeurs des composantes YCbCr d'une image à partir des composantes RVB comme suite :

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.169 & -0.331 & 0.5 \\ 0.5 & -0.81 & -0.81 \end{bmatrix} \times \begin{bmatrix} R \\ G \\ B \end{bmatrix},$$

### \*Signal vidéo

Le procédé mis en œuvre pour reproduire un signal vidéo sur un écran de télévision consiste à afficher une nouvelle image à une fréquence élevée, à la manière d'un film. Cette fréquence est supérieure à 25 Hz, fréquence en dessous de laquelle on percevrait une discontinuité (25 images par secondes).

### \*La compression sans perte

#### Codage par plage RLE (Running Length Encoding)

##### Le Run-length encoding ou RLE

Cette méthode de compression sans perte code les données en y recherchant des successions d'octets identiques. Le RLE est utilisé pour compresser des images codées sur un faible nombre de bits car, dans [ce type d'image](#), il est fréquent de trouver des pixels contigus identiques. La méthode RLE est utilisée pour la compression des images au format [PCX](#), dans une version du format [BMP](#) ainsi qu'au format [JPEG](#).

##### Principe

- Recherche de séquences de données redondantes (ex. niveaux identiques).
- Codage de la valeur et du nombre de répétitions.

##### Exemple :

0	0	0	11	11	67	121	121	98	98	98	32	37	37	37	37	37	37	2	0
↓																			
0	3	11	2	67	1	121	2	98	3	32	1	37	6	2	1	0	1		

**Remarque :** Les images ne contenant que deux couleurs (noir et blanc) sont traitées par des algorithmes de type RLC. C'est le cas des algorithmes utilisés pour la transmission de fax.

#### Le codage de Huffman

Cette technique de compression sans perte repose sur le codage statistique (entropique). Plus les symboles apparaissent souvent, plus les codes qui les remplacent sont courts.

Le codage de Huffman est utilisé pour la compression du son au format [MP3](#) et des images aux formats [PNG](#) et [JPEG](#).

##### Principe de fonctionnement :

L'algorithme parcourt les données d'un fichier et examine chaque octet ainsi que le nombre de fois où ce dernier apparaît.

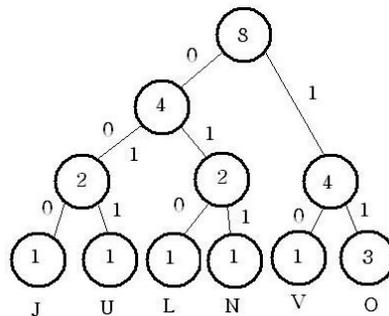
Ensuite, l'algorithme construit un arbre (ou graphe). La construction de l'arbre s'effectue en commençant par ses feuilles jusqu'à sa racine. Pour ce faire :

1. Trier les différentes valeurs par ordre décroissant de fréquence d'apparition.
2. Fusionner les deux poids minimaux dans un arbre binaire et affecter leur somme à la racine.
3. Réordonner la table de poids par poids décroissants.
4. Recommencer en 2. Jusqu'à obtenir un seul arbre.

Construction du code à partir de l'arbre obtenu on attribuant des 1 aux sous-arbres de bas et des 0 à haut.

**exemple :**

Par exemple, le mot "JULONOVO" compte 1 fois les lettres "J, U, L, N, V" et 3 fois la lettre "O".



Les lettres sont donc codées comme suit : J : 000 ; U : 001 ; L : 010 ; N : 011 ; V : 10 ; O : 11. Le caractère "O" qui apparaît le plus souvent reçoit un code plus court que les autres.

**\*Shannon-Fano :**

$a_i$	$p(a_i)$	1	2	3	4	Code
$a_1$	0.36	0	00			00
$a_2$	0.18		01			01
$a_3$	0.18		10			10
$a_4$	0.12	1	110			110
$a_5$	0.09		11	1110		1110
$a_6$	0.07			1111	1111	

**\*La famille des Lempel-Ziv**

Cette famille est un ensemble de techniques de compression sans perte qui utilisent un dictionnaire et où les codes signifient les indices des séquences d'octets placées dans le dictionnaire.

### **LZ77 :**

Le LZ77 forme la méthode de compression utilisée pour compresser les images au format [PNG](#) et, notamment, les données au format [ZIP](#).

### **Principe de fonctionnement :**

LZ77 utilise une fenêtre divisée en deux parties. La partie de droite, de taille plus petite que le reste de la fenêtre, est appelée le tampon et contient les données à compacter. La partie de gauche, appelée fenêtre coulissante ou glissante, peut être assimilée à un dictionnaire. Les octets sont progressivement introduits dans la fenêtre par la droite de celle-ci, c'est à dire en commençant par le tampon.

L'algorithme compare les données à compacter, qui se trouvent dans le tampon, avec les données qui se trouvent dans le reste de la fenêtre. Pour ce faire, il évalue les données à compacter de gauche à droite depuis le début du tampon.

Les séquences des symboles à compacter sont codées par 3 informations  $\langle a, b, c \rangle$ . Si une de ces séquences se trouve déjà dans le reste de la fenêtre alors :

- a signifie la position de la séquence des symboles équivalente qui se trouve dans le reste de la fenêtre par rapport à la séquence des symboles compactée,
- b signifie la longueur de la séquence des symboles compactée,
- c signifie le premier symbole du tampon qui diffère de la séquence des symboles compactée,

Si la séquence d'octets à compacter ne se trouve pas dans le reste de la fenêtre alors  $a=0$  et  $b=0$ . La fenêtre se décale alors de  $b+1$  symboles vers la gauche.

### **Exemple :**

Example: Encode the string "abracadabrrarray" with window size 14 and lookahead buffer size is 6

1	2	3	4	5	6	7	8	1	2	3	4	5	6	Output
								a	b	r	a	c	a	<0,0,a>
							a	b	r	a	c	a	d	<0,0,b>
						a	b	r	a	c	a	d	a	<0,0,r>
					a	b	r	a	c	a	d	a	b	<3,1,c>
			a	b	r	a	c	a	d	a	b	r	a	<2,1,d>
	a	b	r	a	c	a	d	a	b	r	a	r	r	<7,4,r>
c	a	d	a	b	r	a	r	r	a	y				<3,2,y>

← Search Buffer
Look ahead buffer Buffer →

Activate Windows  
 Go to Settings to activate Windows

### \*La compression avec pertes

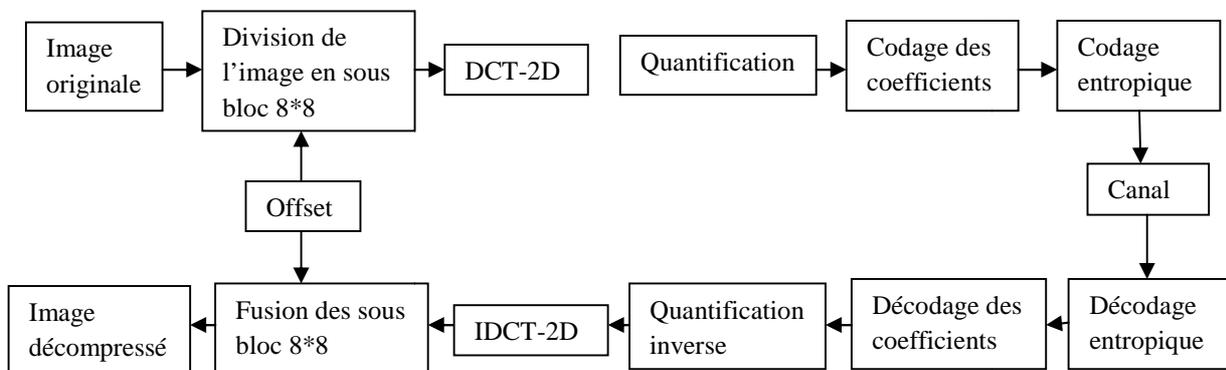
La compression avec pertes est aussi appelée communément compression destructrice. La compression avec pertes utilise des algorithmes qui compressent les données en les dégradant. Autrement dit, après la décompression des données, celles-ci sont moins nombreuses que celles utilisées pour la compression. Pour éviter que ces dégradations soient perçues par les utilisateurs, les concepteurs d'algorithmes pour la compression destructrice se basent sur les limites de perception au niveau de l'ouïe et de la vue chez l'être humain.

Les algorithmes de compression destructrice sont utilisés pour compresser des données graphiques, audio et vidéo. Ils ne pourraient être appliqués sur des données textuelles ou sur celles d'un

programme au risque de les rendre illisibles ou inexécutables.

Les méthodes utilisées pour la compression avec pertes sont **La transformée en cosinus discrète (DCT)** et **La transformée en ondelettes**.

### Compression d'image JPEG (Joint Photographic Experts Group)



Codec (codeur-décodeur) JPEG.

1. L'image est tout d'abord convertie au format  $YCbCr$  pour des raisons d'efficacité de codage.
2. Le signal de luminance (Y) et les signaux de chrominance sont ensuite divisés en blocs de 8x8 pixels.
3. On applique une analyse fréquentielle appelée DCT (Discrete Cosine Transform), dont le principe est similaire à celui de l'analyse de FOURIER, à chaque bloc de 64 pixels. On passe ainsi d'une représentation spatiale de 64 pixels à une représentation fréquentielle avec une composante continue donnant le niveau moyen du bloc.

Formule pour calculer la DCT sur une matrice NxN

$$DCT(i, j) = \frac{1}{\sqrt{2}} C(i) C(j) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \text{pixel}(x, y) \cos\left(\frac{(2x+1)i\pi}{2N}\right) \cos\left(\frac{(2y+1)j\pi}{2N}\right)$$

Formule pour calculer la IDCT sur une matrice NxN

$$\text{pixel}(x, y) = \frac{1}{\sqrt{2N}} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} C(i) C(j) DCT(i, j) \cos\left(\frac{(2x+1)i\pi}{2N}\right) \cos\left(\frac{(2y+1)j\pi}{2N}\right)$$

$$C(x) = \frac{1}{\sqrt{2}} \text{ si } x=0, \text{ et } 1 \text{ si } x>0.$$

4. Les coefficients de la représentation fréquentielle sont quantifiés avec une précision plus faible pour les hautes fréquences étant donné que l'œil ne les perçoit pas.

Grâce à ce facteur de compression, on crée une table de quantification 8 lignes, 8 colonnes :

$$Quantification[i][j] = 1 + (i + j - 1) * qualite;$$

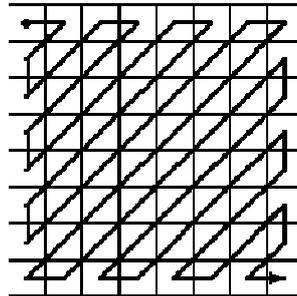
Exemple pour un facteur de qualité de 3 :

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

Ensuite chaque coefficient calculé par la DCT va être quantifié à l'aide de cette table :

$$coefficient[ligne][colonne] = ROUND \left( \frac{coefficient[ligne][colonne]}{Quantification[ligne][colonne]} \right)$$

5-Le codage des coefficients s'effectue en zigzag :



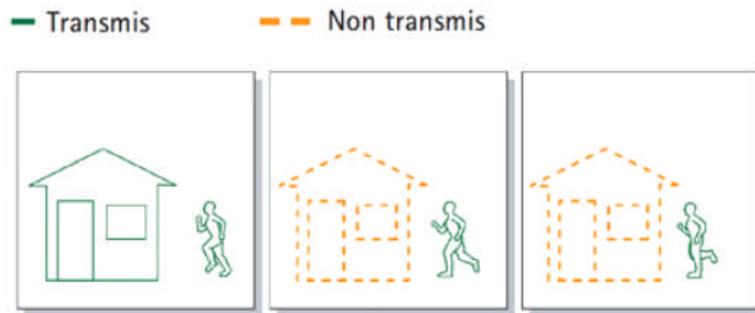
6. Un codage entropique de HUFFMAN (codage sans perte) est finalement appliqué aux coefficients.

### \*Introduction sur la compression vidéo

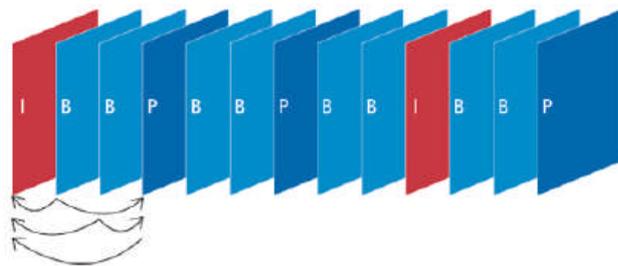
Chaque image est compressée en JPEG



## Un grand nombre d'éléments immobiles dans l'image



- On envoie la différence entre 2 images consécutives



## Cela fait apparaître 3 types d'images

- I pour Intra
- P pour Predit
- B pour Bi-directionnel

### Compression avec perte

« Ne jamais transmettre une donnée déjà transmise »

- **redondance spatiale**
- **redondance temporelle**

### Compression avec perte

« Ne jamais transmettre une donnée déjà transmise »

- Redondance spatial
  - Compression des images
  - On utilise le JPEG
  - Il doit pas être trop élevé

## Compression avec perte

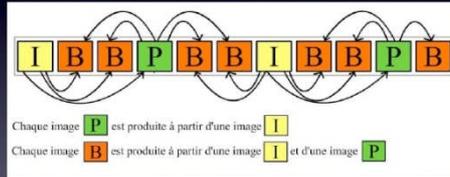
« Ne jamais transmettre une donnée déjà transmise »

- Redondance temporelle
  - On encode que les différences
  - On perd des détails
  - Produit trois types d'images compressés
    - I-frame
    - P-frame
    - B-frame


 GOP : Group of Picture  
 12 à 15 images (MPEG1)

## Compression avec perte

Redondance temporelle

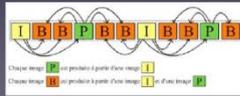


## Compression avec perte

Redondance temporelle

### I - Frame

- Image intracodé : Codé en JPEG
- Il ne dépend pas d'une autre image
- «Image clé» -> KeyFrame

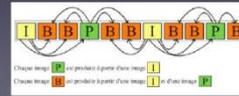


## Compression avec perte

Redondance temporelle

### P - Frame

- Image prédictive
- Contient que les pixels modifiés
- On encode que les différences
- 1/3 taille par rapport à I-Frame



## Compression avec perte

Redondance temporelle

### B - Frame

- Image Bi-directionnelles
- C'est une image prédite à partir de I et P
  - $I(i) + P(i++) = B$
- 1/6 taille I-Frame

